Exploring Cognitive Effects of Inconsistency Characteristics on Understanding Inconsistencies in Declarative Process Models

Sabine Nagel
Institute for IS Research
University of Koblenz
snagel@uni-koblenz.de

Patrick Delfmann
Institute for IS Research
University of Koblenz
delfmann@uni-koblenz.de

Abstract

Inconsistency handling in declarative process models (DPMs) has been of increased interest as even a single contradiction within a constraint set makes the entire DPM unsatisfiable. To develop interactive inconsistency resolution and prevention approaches, humans must be able to identify and understand the problem at hand. Therefore, we aim to gather first insights into the cognitive effects of inconsistency characteristics on understanding inconsistencies in DPMs by conducting an exploratory study. Our results show that participants had notable difficulties with understanding inconsistencies, which we could trace back to several inconsistency characteristics, such as combinations of interrelated constraints. Thus, we identified a strong need for the development of interactive and visual decision support technologies to improve inconsistency understanding in DPMs.

Keywords: Declarative Process Models, Declarative Process Specifications, Declare, Inconsistencies, Comprehension

1. Introduction

Enabling compliance with both internal and external regulations has been identified as a current challenge for organizations (Hashmi et al., 2018). Such compliance regulations are often modeled in the form of declarative statements that define allowed company behavior (Graham, 2006). A common way of modeling company processes using a declarative approach are declarative process specifications (Di Ciccio & Montali, 2022), also referred to as declarative process models (DPMs). DPMs are defined as "a set of constraints that must all be satisfied during the process run" (Figl et al., 2020, p. 123) and accept any execution trace that complies with the defined constraints. However, even a single contradiction within a constraint set makes the entire DPM unsatisfiable, as no finite execution trace is accepted. Thus, handling inconsistencies in DPMs has been of increased interest in previous years (Corea et al., 2019; Corea & Delfmann, 2019; Di Ciccio et al., 2017). DPMs can originate from different sources, such as event logs (Di Ciccio et al., 2017) or natural language text (Lopez et al., 2019). However, current declarative process discovery and extraction approaches do not take potential interrelations between the newly modeled constraints into account, which can lead to contradictory statements within rule sets, also referred to as inconsistencies (Di Ciccio et al., 2017). For example, Corea et al. (2019) were able to show that a DPM mined from real-life event logs with a support factor of 95% led to more than 600 inconsistencies within 207 constraints. Therefore, it is important to be able to resolve and/or prevent such inconsistencies. To date, several approaches for automated inconsistency resolution have been proposed (Corea et al., 2019; Di Ciccio et al., 2017). However, these approaches might not always be plausible in practice, as this might lead to the deletion of potentially business-critical rules. This stresses the need for interactive approaches that include the human in the loop. To be able to successfully identify and resolve inconsistencies, humans must be able to understand that a set of constraints is inconsistent and pinpoint the cause for this inconsistency. So far, inconsistency understanding has only been studied in the context of DMN decision tables (Nagel et al., 2019, 2020), while DPMs have only been studied regarding understanding challenges in general (Figl et al., 2020; Haisjackl et al., 2016). Thus, this work aims to provide first insights into humans' cognitive approaches and perceptions when trying to make sense of inconsistent constraint sets. More specifically, we aim to answer the following research question:

What are the (potential) effects of inconsistency characteristics on understanding inconsistencies in declarative process models?

The results will provide the basis for future studies and support the improvement of inconsistency

URI: https://hdl.handle.net/10125/107107
978-0-9981331-7-1

understanding using business intelligence (BI) in the form of metrics and/or visualizations, which is an important prerequisite for interactive inconsistency handling approaches.

In Section 2 we introduce related work in the areas of DPMs and inconsistencies. Section 3 elaborates on the design of the explorative study conducted in the scope of this work. Section 4 presents and discusses the results of this study. We conclude with a summary and discussion of opportunities for further research.

2. Background & Related Work

2.1. Declarative Process Models

DPMs consist of a set of constraints that every valid process execution must follow (Di Ciccio et al., 2017; Figl et al., 2020), so they represent circumstantial information (Haisjackl et al., 2016). A common language to model declarative processes is called *Declare*, which defines a set of constraint templates (Di Ciccio et al., 2017). These templates are based on linear temporal logic (LTL) and can be used to model constraints without having to be familiar with the underlying formalization (Figl et al., 2020). Table 1 provides an overview of the Declare templates we use as a basis for our study.

Table 1: Overview of Declare Templates

		Template
Existence	Position	Init(a) | End(a)
	Cardinality	ExactlyOne(a) | ExactlyTwo(a) … AtLeastOne(a) | AtLeastTwo(a) … AtMostOne(a) | AtMostTwo(a) … Absence(a)
Relation	Forward	RespondedExistence(a,b) [Chain]Response(a,b)
	Backward	[Chain]Precedence(a,b)
	Coupling	CoExistence(a,b) [Chain]Succession(a,b)
	Negation	Not[Chain]Response(a,b) Not[Chain]Precedence(a,b) Not[Chain]Succession(a,b) NotCoExistence(a,b)

Generally, we distinguish between existence and relation constraints. Existence constraints express restrictions regarding the position or cardinality of a single activity and are automatically activated. In contrast, relation constraints describe the interplay between two activities, with a source activity requiring or prohibiting a target activity. Here, forward relations are activated by their first parameter, backward relations by their second parameter and coupling constraints can be activated in both directions.

In addition to the direction of activation, we also distinguish between undirected (*RespondedExistence and CoExistence*) and directed relations regarding the order of activities in a trace, i.e., an execution sequence. Directed relations can either express that a target activity must eventually (*Response*, *Precedence*, *Succession*) or immediately (*ChainResponse*, *ChainPrecedence*, *ChainSuccession*) follow or precede a source activity and can also be negated.

In our study, we provided participants with an overview of all templates in their textual and visual forms, including their definitions. To avoid redundancy, we refer to Figure 2 and Figure 3 for an overview of natural language descriptions and visual notations for all templates. For a full formalization of all templates and corresponding LTL formulas, we refer to Di Ciccio and Montali (2022).

2.2. Inconsistency in DPMs

A DPM is referred to as inconsistent if it does not accept any finite execution trace (Di Ciccio et al., 2017). Here we distinguish between inconsistencies in the classic-logical sense, which already occur at design time, and potential inconsistencies that only occur during run time (Corea & Delfmann, 2019; Corea & Thimm, 2020). For example, the constraint set {*Init(a), Response(a,b), NotResponse(a,b)*} is classically inconsistent, while the constraint set {*Response(a,b), NotResponse(a,b)*} represents a potential inconsistency and only leads to contradictory conclusions if *a* occurs in a trace. To extract and measure inconsistencies, we apply the notion of minimally inconsistent subsets (MIS) of declarative constraints. MIS are minimal in terms of set inclusion, so the deletion of exactly one element automatically resolves the inconsistency (Corea & Thimm, 2020). We also distinguish between different inconsistency structures. Nagel & Delfmann (2023) have identified a total number of 16 structures, i.e., recurring patterns with shared characteristics that describe how and why an interplay of constraints can lead to inconsistency. More specifically, they distinguish between structures that describe contradictions regarding a fixed or relative **position** in a trace (IS01-IS03), explicit or implicit contradictions regarding the **cardinality** of an activity (IS04-IS09), contradictions regarding the **relation** between two activities, and inconsistencies that occur due to trace **boundaries** (IS14-IS16), i.e., traces having to be finite. Table 2 provides an overview of all structures, corresponding examples are linked in Section 3.1., and for a more detailed explanation, we refer to Nagel & Delfmann (2023).

Table 2: Inconsistency Structures

ID	Structure
IS01	Multiple Start/End Events
IS02	Multiple Direct Predecessors/Successors
IS03	Contradictory Chain
IS04	Explicit Contradictory Cardinality
IS05	Implicit Contradictory Cardinality – Activation
IS06	Implicit Contradictory Cardinality – Single Pair
IS07	Implicit Contradictory Cardinality – Activated Position
IS08	Implicit Contradictory Cardinality – Multiple Pair
IS09	Implicit Contradictory Cardinality – Combined Pair/Activation
IS10	Contradictory Co-Existence
IS11	Explicit Contradictory Order
IS12	Implicit Contradictory Order – Bidirectional Paths
IS13	Implicit Contradictory Order – Single Boundary
IS14	Lack of Space – Local
IS15	Lack of Space – Global
IS16	Loop

2.3. DPM & Inconsistency Understanding

Many works have investigated the understanding of DPMs. In this context, Nagel and Delfmann (2022) conducted a structured literature review and were able to extract seven challenge categories from a total of 19 empirical studies and theoretical discussions on DPM understanding, as summarized in Table 3.

Table 3: DPM Understanding Challenges

Category	Factors/Options
Complexity	size, density, variability, modularization
Individual Constraints	template definitions
Constraint Combinations	pairs of constraints, hidden dependencies
Representation	visual, textual, hybrid, simulation
Reading Order	sequential instead of circumstantial
Background & Experience	affects preferences within other characteristics
Task Type	contextual information, allowed behavior

In contrast, existing works on inconsistency understanding are currently limited to studies that analyze how quantitative measures and visualization techniques affect the understanding of inconsistencies in DMN decision tables (Nagel et al., 2019, 2020). While some of the identified DPM understanding challenges might also apply to inconsistencies, as they are always subsets of DPMs, others might differ due to different characteristics of inconsistencies themselves. For example, only presenting users with one MIS at a time already reduces complexity and is, thus, expected to lower the mental effort required to understand the problem at hand (Nagel & Delfmann, 2022). Another example includes the high degree of connectivity within MIS, which might increase the use of a visual representation, even though this has been found to decrease understanding of entire DPMs. Also, the trade-off between individual characteristics has never been studied. For example, “size has generally been found to increase mental effort; however, large, and rather explicit models might still be easier to understand than small, implicit ones” (Nagel & Delfmann, 2022). Depending on the structure of an inconsistency, its characteristics, and their interplay can be further defined. This includes but is not limited to the type of inconsistency (classic vs. potential), its size, density (ratio between constraints and activities), and variability (the number of different templates).

3. Study

We approach this work in an exploratory manner due to the lack of research and theory regarding inconsistency understanding in DPMs. Here, we are especially interested in investigating similarities and differences compared to existing research on DPM understanding, as there is the possibility that not all understanding challenges align. Furthermore, we aim to identify additional (inconsistency) characteristics and other factors that potentially affect inconsistency understanding. Lastly, we are also interested in possibilities for understanding improvement. In the following, we describe the design, the individual steps, and the data collected during these steps (cf. Figure 1).

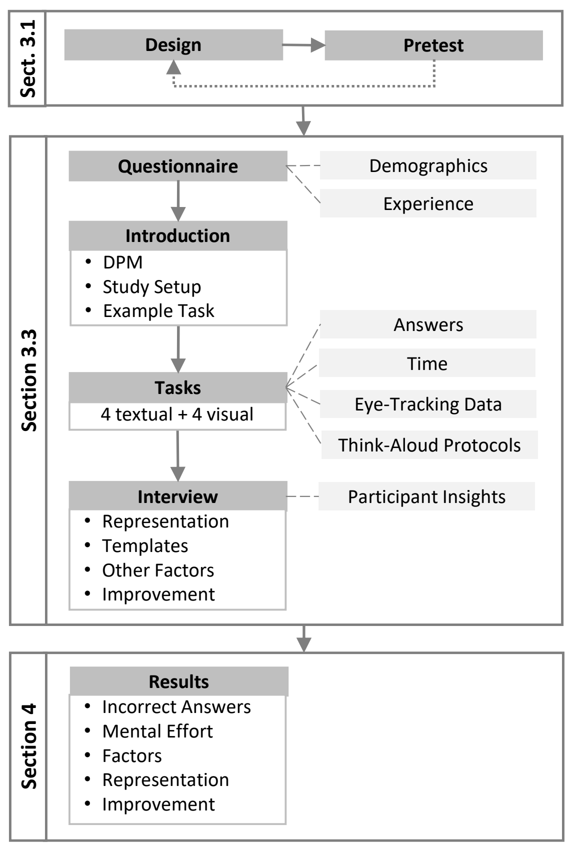

Figure 1: Overview of Study

3.1. Preparation

In this study, we provided participants with constraint sets and asked them to provide a valid trace or explain the underlying problem. Each participant received different constraint sets to collect data for a large variety of inconsistency characteristics. Therefore, we created between two and eight constraint sets based on each inconsistency structure (cf. Section 2.2). To keep the task equal throughout the study, we only used classic inconsistencies, as potential inconsistencies allow valid traces unless the activating activity occurs. We created MIS with randomly generated characteristics for each dimension (e.g., size, template variability) to represent a large variety of different inconsistencies. This led us to a total number of 60 different MIS, which serve as the basis for our study. We are also interested in investigating any differences in inconsistency understanding when confronting participants with textual vs. visual constraint sets. Thus, we prepared each task in textual and graphical representation, leading to a total number of 120 tasks[1].

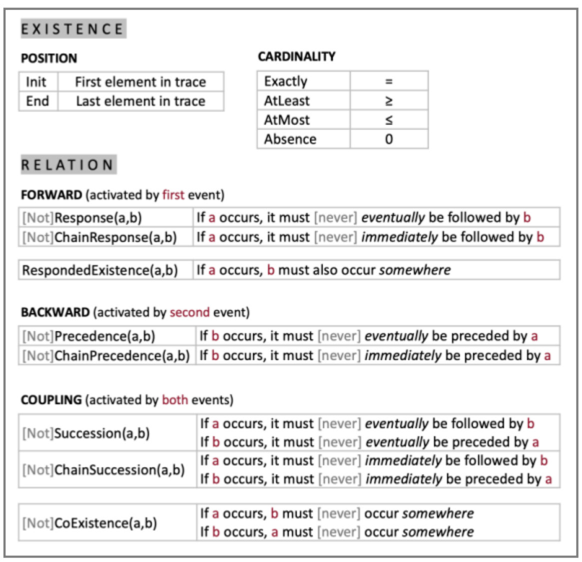

Figure 2: Overview of Textual Constraints

To focus on the interplay between constraints as opposed to the contents of these rules, we used randomly generated letters from *c* to *z* to represent activities in each task. The letters *a* and *b* were excluded from the constraint sets to avoid confusion, as these letters were already used in the provided template overview (Figure 2 and Figure 3).

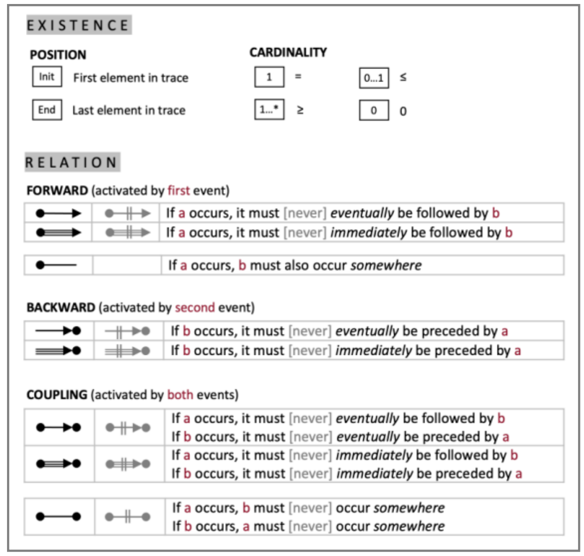

Figure 3: Overview of Visual Constraints

As shown in Figure 4, textual models were structured equally to the corresponding legend and within each sub-category, the constraints were shown in random order. For visual constraints, we randomly selected the direction of each arrow to prevent any bias by using a predefined visualization pattern and ensuring the same conditions for all participants, as they all received different tasks. We also prepared introductory material to provide the participants with all relevant prerequisites and prior knowledge.

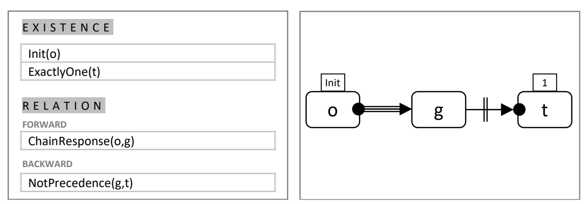

Figure 4: Exemplary Task in Textual (left) and Visual (right) Representation

To verify and iteratively improve our study design and setup, we conducted pretests with the following results. The introduction was revised multiple times to break down the key foundations as briefly as possible, while still covering all concepts required for solving the tasks. Additionally, a step-by-step example was added to the introduction after the second iteration of pretests to prepare the participants for the following tasks. This was necessary due to the participants' lack of experience with DPMs. We also decreased the maximum size of constraint sets from 15 to 8 constraints, as participants were not able to work with

[1]https://uni-ko.de/hicss-iu-study

larger constraint sets in a reasonable amount of time. The constraint overviews were also iteratively refined to improve their usability.

3.2. Participants & Settings

The study was conducted in November of 2022 with 20 undergraduate and graduate students (cf. Table 4) from the faculty of computer science at the University of Koblenz. While this sample size is generally considered small, it is a common sample size for exploratory studies in the area of DPMs "due to the substantial effort to be invested per subject" (Haisjackl et al., 2016, p. 330). All students had basic prior knowledge of procedural business process modeling and logic/reasoning, which we verified as part of our pre-study questionnaire. To ensure equal prerequisites, no prior knowledge of DPMs was allowed. This was important to prevent distortion of our results, as prior experience with certain templates or inconsistencies in DPMs might influence inconsistency understanding. Participation in the study was voluntary and no incentive was offered.

We created and executed our study using the cloud-based eye-tracking software EYEVIDO Lab and a Tobii 4C eye-tracker. The study was displayed on a 24-inch screen with a resolution of 1920 x 1080. The tasks did not require or allow scrolling or zooming, as all components were shown at once, and clear visibility from a viewing distance of 60 cm was verified as part of our pretest (cf. Section 3.1). To keep the settings the same for all participants, we conducted the study in our IT lab where we could fully control potential distractions such as noise and lighting.

We added our introductory slides as a PDF file and implemented the tasks as an HTML file. This allowed us to easily replace the respective models after each run without having to edit the study itself. Furthermore, it allowed us to easily track the participant's answers. All other measurements were provided by EVEVIDO Lab and are explained in more detail in the following section.

3.3. Structure, Instrumentation & Measurements

The data collection of our study was structured as follows. First, we welcomed the participants and asked them to fill out a pre-study questionnaire and consent form. Here, we asked participants to provide their program of study and rate their prior knowledge in the areas of process modeling and logic/reasoning using a 5-point Likert scale. This was followed by an explanation of the study and the calibration of the eye tracker. Next, the participants were provided with introductory slides to read at their own speed.

As part of the data collection, each participant was confronted with a total number of eight constraint sets. Six were minimally inconsistent, while the remaining two were similar consistent constraint sets (i.e., we removed a single constraint from an MIS to make it consistent) that serve as control questions.

The screen was divided into three areas for each question. The question area was located at the top of the screen. The task was the same for all constraint sets ("Please provide a valid trace for the model below.") and the participants could either provide a valid trace and press "Submit" or conclude that there is no valid trace by pressing "No valid trace". Below, the textual or visual constraints were located on the left (cf. Figure 4), while an overview of all templates (Figure 2 and Figure 3) was located on the right. After submitting a valid trace, the participants were automatically directed to the next question. In case the participants concluded that there was no valid trace, they were directed to a follow-up screen, which consisted of the same three areas. However, the task was replaced with a request to "Please explain in your own words, why there is no valid trace for the given model". After verbally describing the problem, the participants could proceed to the next question. Generally, we assume successful comprehension if a participant has determined that there is no valid trace for a given MIS and was able to correctly explain the problem afterward. The latter was important to identify invalid answers. For all inconsistencies that were successfully understood, we measured understanding efficiency by tracking the time from the point where the task was displayed until the participant clicked on the "no valid trace" button. After completing all tasks, we conducted a short semi-structured interview with the participants to gain further subjective insights into their approach and perception. Here, we asked the following four questions: (1) "Which representation did you find easier to understand and/or more intuitive?", (2) "What templates did you find easy or hard to understand and apply?", (3) "Which other factors or model characteristics did you find to positively or negatively influence your inconsistency understanding?", and (4) "How could inconsistency understanding be improved in the future?". Depending on the answers, we asked follow-up questions for clarification purposes. To prevent bias, we did not disclose task results before study completion.

To gain insights into the cognitive processes of the subjects we evaluated the collected eye-tracking data. Here, we analyzed the fixation duration (FD), i.e., the time the eyes remain still on a fixed location, as this is an indicator of objective mental effort

(Meghanathan et al., 2015), with a higher FD indicating an increased mental effort. Additionally, we considered the number of fixations within different areas of interest (AOI) and evaluated heatmaps and view paths to get further insights on the participants' approaches. To enrich these results with subjective participant insights, we made use of think-aloud protocols during the study (Ericsson & Simon, 1980). We encouraged German-speaking participants to voice their thoughts in German instead of English if that made them more comfortable to speak.

4. Results & Discussion

We processed and analyzed our data as follows. First, we determined which tasks were answered correctly, incorrectly, or were considered invalid. Furthermore, we transcribed the verbal data collected during the study and the post-study interview. We also analyzed the collected eye-tracking data and screen recordings to identify the cause for incorrect answers in case this could not be derived from the think-aloud data and answers. After analyzing all data, we were left with a total of 66 correct (white), 38 incorrect (red), and 16 invalid answers (gray) for non-control questions.

Table 4: Overview of Participants & Results

P	M	L	Intro (min)	Part 1				Part 2			
				1	2	3	4	5	6	7	8
P1	1	3	15:43	5	15	C	24	31	40	C	53
P2	4	2	12:29	1	11	C	29	33	43	C	60
P3	3	4	14:39	2	12	C	30	38	46	C	51
P4	2	2	14:13	3	20	C	32	41	50	C	56
P5	1	4	23:08	4	22	C	35	42	45	C	57
P6	2	1	17:04	6	13	C	18	26	48	C	49
P7	1	4	11:40	7	14	C	16	28	44	C	52
P8	2	1	15:50	8	17	C	23	37	39	C	54
P9	2	3	15:11	9	19	C	25	34	47	C	58
P10	2	4	15:54	10	21	C	27	36	55	C	59
P11	3	2	11:54	53	40	C	31	24	15	C	5
P12	2	4	13:10	60	43	C	33	29	11	C	1
P13	2	4	21:19	51	46	C	38	30	12	C	2
P14	3	4	15:26	56	50	C	41	32	20	C	3
P15	3	3	11:11	57	45	C	42	35	22	C	4
P16	4	4	15:51	49	48	C	26	18	13	C	6
P17	4	2	14:28	52	44	C	28	16	14	C	7
P18	2	2	19:41	54	39	C	37	23	17	C	8
P19	3	2	11:06	58	47	C	34	25	19	C	9
P20	1	3	21:56	59	55	C	36	27	21	C	10

Table 4 provides an overview of the data that serves as the basis for the following analyses. Here we show the participant IDs, their prior knowledge in the areas of process modeling (M) and logic/reasoning (L), the time they spent looking at the introductory slides, the task IDs for each participant, and their result for each task, indicated by the previously described colors. While odd participant IDs received textual models in part 1 and visual models in part 2, the order was reversed for participants with even IDs.

In the following sections, we first identify the reasons for tasks being answered incorrectly, analyze the obtained eye-tracking data to gain insights into the objective mental effort, and then discuss the perceived mental effort in more detail.

4.1. Reasons for Incorrect Answers

In the introduction, we instructed participants to start with the existence constraints and follow the path of activation from there, as relation constraints represent if-statements that must be activated by an activity. We also suggested using the input field to start constructing traces while analyzing the constraints. Despite these instructions, we observed a variety of different approaches that often led to problems when trying to make sense of the provided constraint sets. While some incorrect answers were based on a single error, others resulted from a sequence of errors. Table 5 provides an overview of the main reasons for incorrect answers, as well as the corresponding number of affected textual and visual tasks, as discussed in more detail below.

Table 5: Overview of Reasons for Incorrect Answers and Number of Tasks

Reason	Textual	Visual
Constraints were skipped	6	7
Constraints were not applied globally	4	2
Inactivated relations were executed	4	4
Templates were misunderstood/misinterpreted	21	16
Other	2	6

Constraints were skipped for several reasons, mostly because the activation of a relation constraint was missed, or constraints were simply overlooked or not considered at all. Especially the latter was a larger problem for visual constraints (4 vs. 1 incorrect answers), as participants had an immediate overview of all interrelations but quickly jumped to the conclusion that parts of the model were irrelevant. Another commonly made mistake was not considering relations as if-statements and adding all involved activities to the trace, which led to unnecessary and redundant activities.

Next, some constraints were not applied to the entire trace, which was a larger problem for textual models. For example, one participant provided the combination "cgcf" as a valid trace for the constraint set {*AtLeastTwo(c)*, *ChainResponse(c,g)*, *ChainResponse*(*c*,*f*)}. However, both *ChainResponse* constraints were only applied to a single occurrence of *c*, which led to an incorrect answer. It also occurred that conditions were initially applied correctly but were never revisited after updating the trace.

The most common causes for incorrect answers were misinterpreted or misunderstood templates. While forward templates were rather unproblematic, backward constraints were regularly misinterpreted. More specifically, *Precedence* was either confused with *Response* ("*a* must be followed by *b*" instead of "*b* must be preceded by *a*") or the parameters themselves were switched ("*a* must be preceded by *b*"). Here the visual representation appears to prevent this issue compared to a textual representation, with 1 vs. 6 incorrect answers. Furthermore, regular constraints were sometimes confused with their chain counterpart and vice versa, positive and negative templates were mixed up, and cardinalities were processed incorrectly. In Section 4.3 we will extend this discussion by focusing on the participants' perception regarding templates.

Other errors include not fully grasping the concept of a finite trace (i.e., trying to provide an infinite trace or being confused by parallel activity executions), automatically considering the "last" activity in visual representations the end, or interpreting multiple cardinality constraints involving a single activity as alternatives.

4.2. Understanding Accuracy & Objective Mental Effort

When looking at the distribution of correct answers across tasks, some inconsistency structures generally seem to be easier to understand than others (see Table 6). This can mainly be explained by their complexity (i.e., size and template variability). For example, tasks containing directly contradicting constraints (e.g., IS01, IS03) were all answered correctly, while participants struggled with more complex and interrelated structures (e.g., IS09). While the overall distribution was similar with 6 structures having more correct answers for textual vs. 7 structures for visual constraints, some structures seem to explicitly benefit from a visual representation (e.g., loops) and vice versa.

When looking at the average fixation duration, it is noticeable that the visual notation is associated with a lower mental effort required to understand an inconsistency for most structures (11 out of 16).

Table 6: Overview of Correctly Answered Questions

ID	Tasks	Correct Answers		Fixation Duration (s)	
		Textual	Visual	Textual	Visual
IS01	1–2	100%	100%	98	10
IS02	3–6	25%	50%	61	160
IS03	7–10	100%	100%	119	66
IS04	49–52	100%	75%	28	21
IS05	57–60	75%	25%	89	125
IS06	53–56	0%	50%	144	80
IS07	45-46	0%	50%	88	69
IS08	43–44	100%	50%	102	87
IS09	47–48	50%	0%	157	110
IS10	31–34	100%	50%	85	172
IS11	11–14, 35–38	25%	38%	169	139
IS12	19–22	25%	50%	200	99
IS13	15–18	75%	25%	67	101
IS14	23–26	75%	75%	200	81
IS15	27–30	75%	100%	77	20
IS16	29–42	0%	50%	136	166

Next, we considered other characteristics that have the potential to affect inconsistency understanding (cf. Section 2.3). While most characteristics strongly influence each other and can, therefore, not be considered in isolation, we were able to observe a clear trend regarding the number of constraints and the mental effort required to understand the respective constraint sets. As shown in Table 7, the average fixation duration for tasks involving constraint sets of increasing size also increases, regardless of most other characteristics. This aligns with related works on DPM understanding (cf. Section 2.3 and Abbad-Andaloussi et al. (2023)). Also, we can again see a difference between the mental effort required to understand inconsistencies in visual models and textual inconsistencies. While this is also the case when considering the understanding accuracy, which is measured using the number of correct answers, a clear trend of a decreased understanding accuracy for increasing inconsistency size cannot be observed.

Table 7: Average Fixation Duration for Tasks with Increasing Numbers of Constraints

Constraints	Correct Answers		Fixation Duration (s)	
	Textual	Visual	Textual	Visual
2	88%	88%	46	40
3	40%	60%	61	82
4	25%	25%	96	92
5	40%	50%	145	106
6	30%	30%	134	107
7	25%	50%	159	155
8	44%	67%	192	142

4.3. Perception of Factors Affecting Inconsistency Understanding

Based on the collected think-aloud protocols and the answers resulting from our post-study interview, we now investigate and discuss the perceived mental effort when trying to understand inconsistencies in DPMs.

While only a few participants perceived the templates themselves as easy to understand and rather straightforward (P7, P19), most participants had notable difficulties with one or more templates. Existence constraints were mainly considered as easy to understand (P7, P8) and especially cardinality was mentioned as an intuitive concept known from other areas (P11, P17). 4 participants even implied that the presence of an Init constraint helped them to understand the model "as you always knew where to start" (P3). For relation constraints, it was agreed that backward constraints are harder to understand and less intuitive than forward or coupling constraints, mainly because of their reversed order of activation. Negations were generally perceived as easy to understand, independent of the form of representation, except requiring "twice the mental effort to apply" (P5). Chain constraints were also considered easy to implement, "as the condition that two activities must follow each other directly is satisfied right away and you do not have to go back" (P19), the latter being the case for regular relations.

Another factor that was mentioned by participants is the complexity of an inconsistency. Here, 11 participants agreed that a larger number of constraints negatively influenced inconsistency understanding, while some participants specifically mentioned the number of relation constraints or activities.

Furthermore, two participants implied that an increased template variety makes it harder to understand the provided models. For example, P5 referred to the number of different symbols in visual constraint sets as "an increased diversity makes the logic very complex" and P11 specifically said that "when there was a bit of everything, it was harder".

Furthermore, the participants mentioned several factors related to individual constraints or constraint combinations. Two participants were confused by an activity that was not connected to the remaining activities (P4, P6). While P16 and P17 generally referred to the presence of interrelations as problematic, others specifically mentioned combinations of forward and backward constraints (P13, P17) and/or multiple arrows connected to a single element (P6, P7, P20). Also, P8 mentioned that "it is harder if rules that contradict each other are further apart". However, as inconsistencies are minimal, all constraints are automatically involved in the contradiction. Lastly, direct contradictions made it easier for the participants to identify problematic models (P4, P7, P14), compared to contradictions that are hidden across multiple constraints.

4.4. Textual vs. Visual Representation

Table 8 provides a summary of the number of correctly answered questions, the percentage of fixations on the screen area containing the provided legend, and the participants' perception regarding which form of constraint representation they found easier. In total, 11 participants preferred the textual representation while 8 preferred the visual notation. This shows that there does not seem to be a unanimous favorite, but the preferred notation is based on personal preference. Interestingly, the participants' perception does not always align with the number of correctly answered textual and visual tasks, as well as the mental effort dedicated to the respective legends. Also, multiple participants expected to prefer the visual notation after the introduction and later concluded, that they found the textual constraints easier to understand (P1, P6, P10, P12). Additionally, P3 and P6 implied that they imagined visual constraints to be easier after having some practice, which they were lacking during this study. We now discuss the main advantages and disadvantages mentioned by the participants.

Table 8: Textual vs. Visual Constraint Representation

PID	Correct		Legend Fixations (%)		Perception
	Text.	Vis.	Textual	Visual	
P1	2	2	30.3	31.3	textual
P2	3	4	20.0	6.4	visual
P3	4	3	24.3	38.6	textual
P4	1	2	19.4	38.0	textual
P5	0	2	17.4	17.1	visual
P6	1	0	23.4	32.6	textual
P7	3	3	24.2	14.2	visual
P8	0	4	21.4	30.7	textual
P9	2	1	31.7	31.6	textual
P10	1	3	19.1	12.8	textual
P11	1	2	36.1	37.3	visual
P12	3	0	32.0	43.4	textual
P13	1	2	23.1	24.6	visual
P14	3	3	22.9	9.6	same
P15	1	3	36.8	28.0	textual
P16	0	0	20.1	17.6	textual
P17	3	1	25.8	35.2	textual
P18	4	4	20.8	21.0	visual
P19	3	2	32.4	28.6	visual
P20	1	0	28.2	31.4	visual

Multiple participants agreed that textual constraints require more time to read and process (P5, P6, P11), while visual constraint sets are easier to look at and faster to process (P11, P19, P20). Also, visual constraints allow seeing the big picture as they display dependencies and interrelations between constraints (P2, P3, P5, P19), while textual constraints require any context to be stored in the participants' minds (P7). However, this also leads to having to process more information at a time when looking at visual models (P14, P16, P17), whereas textual constraints can be read one at a time (P8). Thus, a visual notation seems especially beneficial for smaller examples (P1, P14).

As MIS only comprise a small fraction of constraints compared to the overall model, a visual representation might be more suitable for improving inconsistency understanding compared to the understanding of DPMs in general (cf. Section 2.3).

The interrelations between visual constraints also helped participants to follow a path of activation (P3, P6) whereas activity occurrences had to be identified manually to put textual constraints in a logical order (P3, P6, P13). Considering that MIS have a high degree of connectivity, this seems to be an important factor when presenting inconsistencies to users. However, multiple participants also referred to the visual notation as confusing, abstract, and unintuitive. Here, some complained about the arrows, especially for backward constraints (P4, P15), while others found the circles that represent activations hard to understand (P7). In contrast, one advantage of textual constraints appeared to be the rather self-explanatory wording (P1, P7). This aligns with the opinions about prior experience helping to understand textual and/or visual constraints. While some found the visual notation easier "as it is similar to common graph notation" and they could "associate symbols with known ones" (P7), others criticized the similarity of the visual models to procedural process models, which led to confusion (P9) as it might, e.g., mistakenly imply the presence of loops (P10).

To summarize, a visual notation seems preferable when providing rather small and strongly connected constraint sets, but the notion might require some adjustments to improve understanding, which we will discuss in more detail in the following section.

4.5. Improving Inconsistency Understanding

To improve inconsistency understanding, many users suggested changing the order in which the constraints are displayed. For textual constraints, it was suggested to display the constraints in their order of activation (P11), although that might not be possible in many cases due to multiple activation paths. For visual constraints, many participants struggled with not having a clear start (P11) and reading order, which is due to the nature of DPMs comprising circumstantial and not sequential information. However, P8, P11, P14, and P18 suggested displaying all constraints with the arrows pointing from left to right or top to bottom, as this resembles their natural reading order. While this is generally possible, it would mean that the order of activation might still be visualized from right to left or bottom to top. Some also realized this issue and only suggested changing the order, without having any specific visualization in mind. Similarly, two participants suggested completely changing the visual notation, but they were also not able to provide any specific suggestions on how a different visualization could look.

Next, some participants suggested applying color codes to differentiate between templates or groups of constraints. This includes using different colors for different arrows (P12), specifically highlighting negation constraints in red (P17, P18, P20) or coloring *Chain* constraints (P8, P20). Other suggestions for improvement include to "not show as much at once when you have to look for a valid trace but show everything at once when you look for inconsistencies" (P9), spacing out activities visually for large models (P20), and to "clearly mark optional activities" (P10).

Lastly, one participant suggested using a combination of textual and visual representation, which "might make it slower but lead to more correct answers" (P4). This seems like a reasonable suggestion, considering that we were able to show that there is no common understanding about which form of representation is easier or harder to understand, as both textual and visual constraint sets have advantages and disadvantages.

5. Conclusion

In this work, we conducted an exploratory study to investigate the potential effects of inconsistency characteristics on inconsistency understanding in DPMs. We confronted participants with declarative constraint sets and asked them to provide any valid trace or explain the underlying problem. Our results show that many participants had problems with identifying and properly explaining inconsistencies. Thus, we focused on analyzing the causes by looking at the reasons for incorrectly answered questions based on the verbal transcripts, answers, and collected eye-tracking data. We could show that the main characteristic of inconsistencies, namely being a set of interrelated constraints, had negative effects on understanding. Also, backward templates posed notable understanding challenges. Furthermore, we gained subjective insights by conducting post-study interviews with all participants. Here our focus was to identify possibilities for the improvement of inconsistency understanding. The findings include matching the order of constraints to a natural human reading order, as well as making use of color codes for different concepts within inconsistent constraint sets.

However, we found that many participants only identified problems but were unable to come up with a solution, which highlights the non-triviality of an optimal form of representing inconsistencies. Therefore, further studies on the visualization of inconsistencies in DPMs are needed. As several attempts to develop alternative visual representations of DPMs have been made (Ferro & Marrella, 2018; Hanser et al., 2016), these works could serve as a starting point by investigating the potential of these notations in the scope of inconsistency understanding.

To ensure internal validity and prevent distortion of our results, we enforced equal prerequisites by only considering participants without prior knowledge of DPMs. However, the high number of incorrect answers and identified reasons indicate that participants had notable difficulties when working with the declarative notation itself. In future work, we plan to conduct further studies with larger sample sizes and more experienced modelers to decrease the cognitive load of acquiring a new notation and potentially obtain more accurate results.

As this study is exploratory, our results do not point to causal relationships between inconsistency characteristics and the accuracy and mental effort required for understanding inconsistencies. Instead, we focus on gaining first insights into humans' perception and cognitive processes when trying to make sense of MIS. As this is the first work to empirically investigate inconsistency understanding in DPMs, it provides an important foundation for future quantitative studies and the design, development, and evaluation of novel decision support technologies.

6. References

Abbad-Andaloussi, A., Burattin, A., Slaats, T., Kindler, E., & Weber, B. (2023). Complexity in declarative process models: Metrics and multi-modal assessment of cognitive load. *Expert Systems with Applications*, *233*.

Corea, C., Deisen, M., & Delfmann, P. (2019). Resolving Inconsistencies in Declarative Process Models based on Culpability Measurement. *In Proceedings Der 14. Internationalen Tagung Der Wirtschaftsinformatik*.

Corea, C., & Delfmann, P. (2019). Quasi-Inconsistency in Declarative Process Models. *Proceedings of the Business Process Management Forum Co-Located with the 17th International Conference on Business Process Management (BPM 2019)*, 20–35.

Corea, C., & Thimm, M. (2020). On quasi-inconsistency and its complexity. *Artificial Intelligence*, *284*, 103276.

Di Ciccio, C., Maggi, F. M., Montali, M., & Mendling, J. (2017). Resolving Inconsistencies and Redundancies in Declarative Process Models. *Information Systems*, *64*, 425–446.

Di Ciccio, C., & Montali, M. (2022). Declarative Process Specifications: Reasoning, Discovery, Monitoring. In W. M. P. van der Aalst & J. Carmona (Eds.), *Process Mining Handbook* (Vol. 448, pp. 108–152). Springer International Publishing.

Ericsson, K. A., & Simon, H. A. (1980). Verbal Reports as Data. *Psychological Review*, *87*(3), 215–251.

Ferro, L. S., & Marrella, A. (2018). VERTO: A visual notation for declarative process models. *Proceedings of the 2018 International Conference on Advanced Visual Interfaces*, 1–3.

Figl, K., Di Ciccio, C., & Reijers, H. A. (2020). Do Declarative Process Models Help to Reduce Cognitive Biases Related to Business Rules? *Proceedings of the International Conference on Conceptual Modeling*, *12400*, 119–133.

Graham, I. (2006). *Business rules management and service oriented architecture: A pattern language*. John Wiley.

Haisjackl, C., Barba, I., Zugal, S., Soffer, P., Hadar, I., Reichert, M., Pinggera, J., & Weber, B. (2016). Understanding Declare models: Strategies, pitfalls, empirical results. *Software & Systems Modeling*, *15*(2), 325–352.

Hanser, M., Ciccio, C. D., & Mendling, J. (2016). A New Notational Framework for Declarative Process Modeling. *Softwaretechnik-Trends*, *36*(2).

Hashmi, M., Governatori, G., Lam, H.-P., & Wynn, M. T. (2018). Are we done with business process compliance: State of the art and challenges ahead. *Knowledge and Information Systems*, *57*(1), 79–133.

Lopez, H. A., Marquard, M., Muttenthaler, L., & Stromsted, R. (2019). Assisted Declarative Process Creation from Natural Language Descriptions. *2019 IEEE 23rd International Enterprise Distributed Object Computing Workshop (EDOCW)*, 96–99.

Meghanathan, R. N., van Leeuwen, C., & Nikolaev, A. R. (2015). Fixation duration surpasses pupil size as a measure of memory load in free viewing. *Frontiers in Human Neuroscience*, *8*(1063).

Nagel, S., Corea, C., & Delfmann, P. (2019). Effects of Quantitative Measures on Understanding Inconsistencies in Business Rules. *Proceedings of the 52nd Hawaii International Conference on System Sciences (HICSS 2019)*, 146–155.

Nagel, S., Corea, C., & Delfmann, P. (2020). Cognitive Effects of Visualization Techniques for Inconsistency Metrics on Monitoring Data-Intensive Processes. *Information Systems Management*, 1–16.

Nagel, S., & Delfmann, P. (2022). Investigating Inconsistency Understanding to Support Interactive Inconsistency Resolution in Declarative Process Models. *ECIS 2022 Research-in-Progress Papers*.

Nagel, S., & Delfmann, P. (2023). Identification, Abstraction and Classification of Inconsistency Structures in Declarative Process Models. *ECIS 2023 Research Papers*.