

Kathrin Figl & Claudio Di Ciccio & Hajo Reijers



Do declarative process models help to reduce cognitive biases related to business rules?

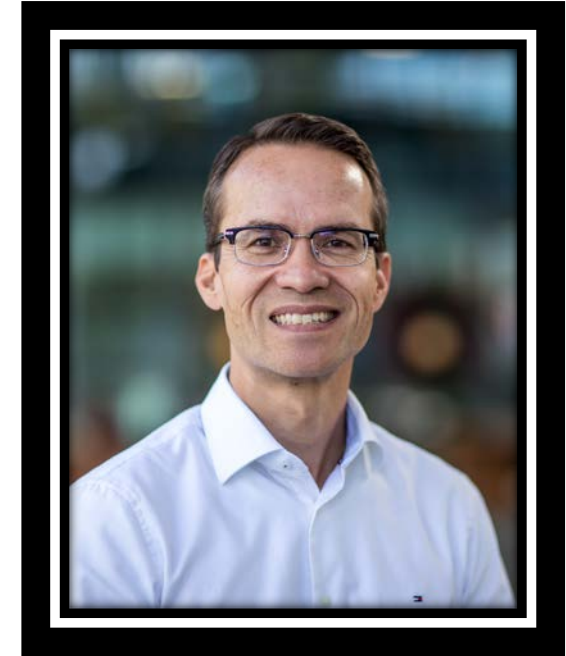
The Authors



Kathrin Figl



Claudio Di Ciccio



Hajo Reijers



SAPIENZA
UNIVERSITÀ DI ROMA

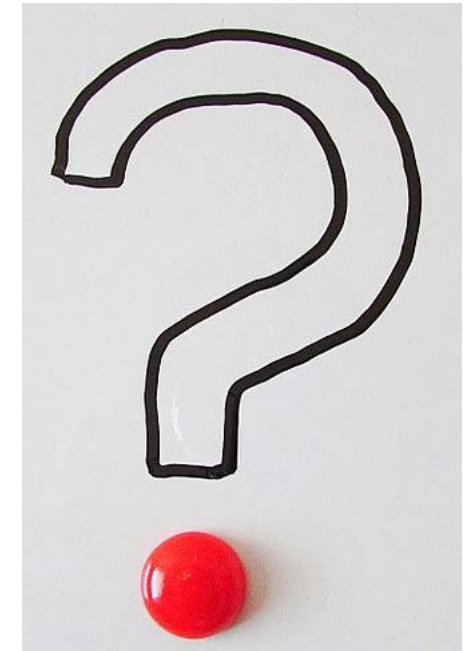


Universiteit Utrecht

Kathrin Figl & Claudio Di Ciccio & Hajo Reijers

Introduction

- » Declarative process modeling languages, such as DECLARE, represent processes by means of temporal rules, namely constraints.
- » The understandability of declarative process models is still a matter of debate.
- » **Research question:** Do supplementary DECLARE models help novice users to understand textual descriptions of business rules better?



Business Rules

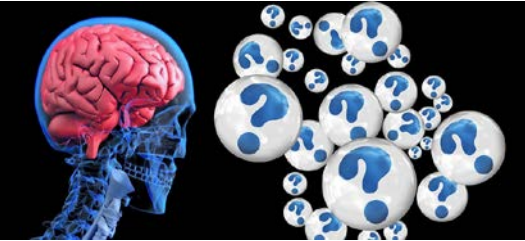


Documentation of business rules is relevant to make them transparent and to avoid rule conflicts.

As business rules can help organizations to achieve their goals, e.g., by reducing costs or improving communication, their proper understanding by all human actors involved is crucial.

In practice, when using natural language to document business rules, conditional if-then statements (if *cause*, then *effect*) are made to describe causal relationships.

In this paper, we focus on the expression of business rules in natural language and in declarative process models.



Deductive Reasoning

“Natural” human reasoning may not always be sound. Humans are prone to typical misinterpretations of if-then statements. Example of the four standard conditional inferences based on a business rule according to formal logics:

	Affirmative	Negative
Valid	If a rental car is returned late, then a penalty is charged. The rental car is returned late. Therefore, a penalty was charged. “Modus ponens”	If a rental car is returned late, then a penalty is charged. A penalty is not charged. Therefore, the rental car was not returned late. “Modus tollens”
Invalid	If a rental car is returned late, then a penalty is charged. A penalty is charged. Therefore, the rental car was returned late. “Affirmation of the consequent”	If a rental car is returned late, then a penalty is charged. The rental car is not returned late. Therefore, a penalty was not charged. “Denial of the antecedent”

Logical Fallacies

Humans are prone to typical misinterpretations of if-then statements and logical fallacies.

Example premise: “If it’s raining then the streets are wet.”

The commutation of conditionals “If the streets are wet then it’s raining” would be logically incorrect, but people are still likely to make this logical error, because in reality it might be a good rule of thumb.

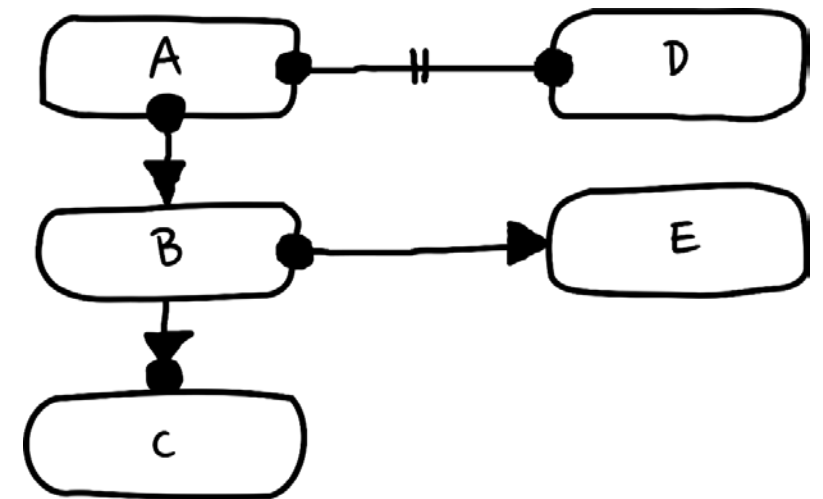


DECLARE

Declarative process models define the behavior of a process by means of constraints, i.e., temporal rules that specify the conditions under which activities may, must, or cannot be executed.

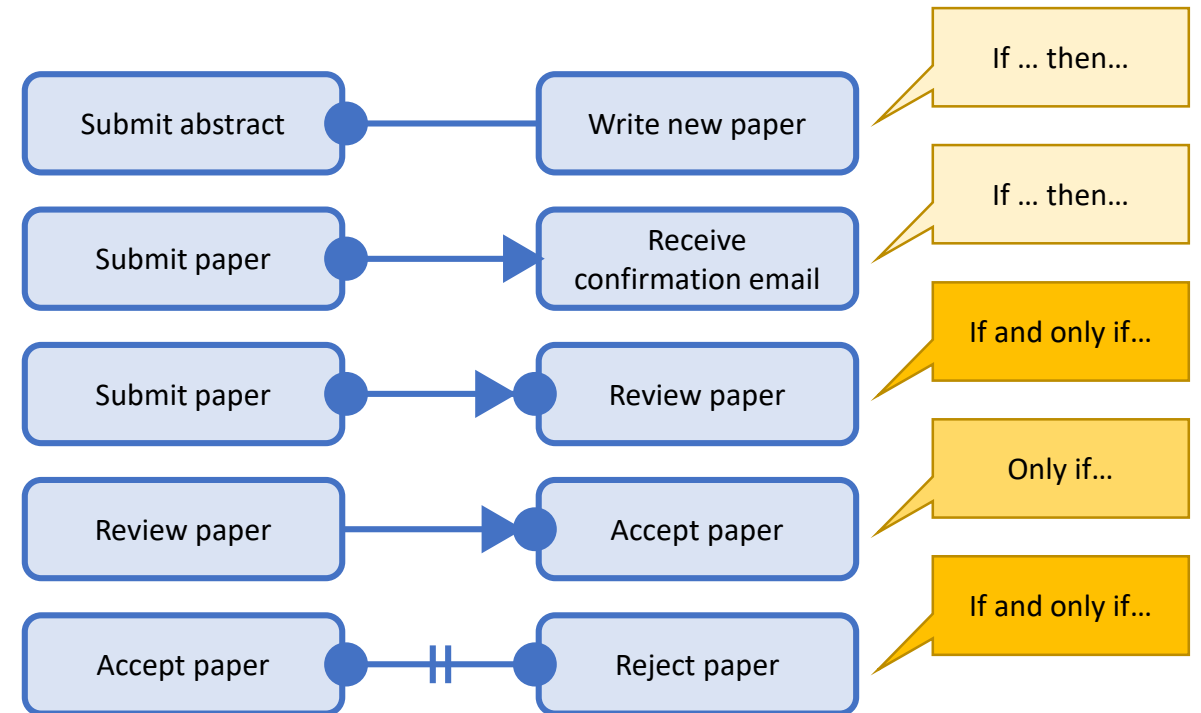
A well-known declarative process modeling language is DECLARE.

DECLARE defines a repertoire of rule templates.



DECLARE: an Example

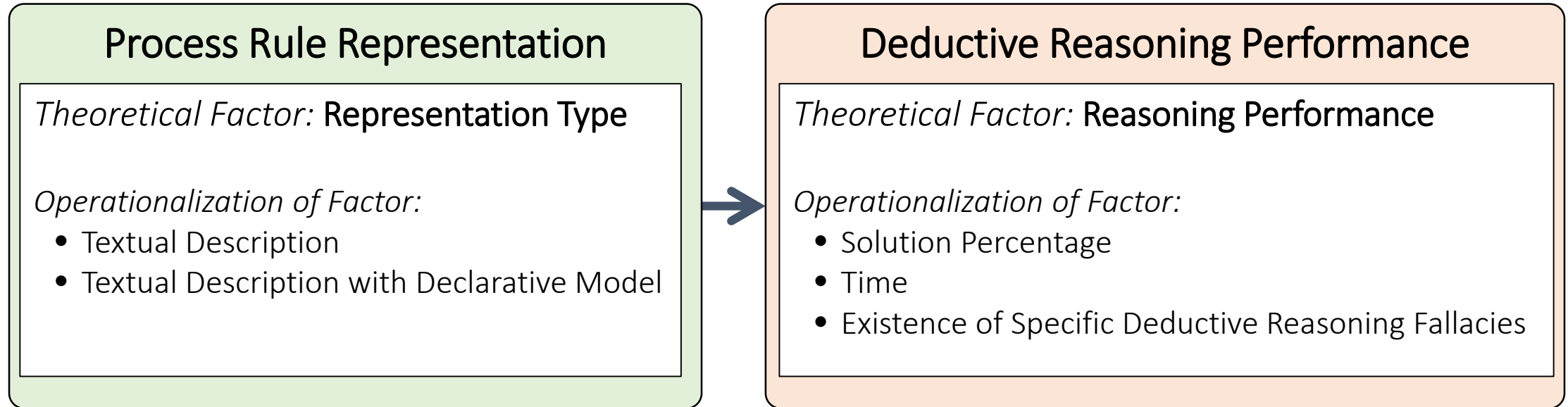
- If an abstract is submitted, a new paper had been or will be written
RespondedExistence(Submit abstract, Write new paper)
- After the paper submission, a confirmation email is received
Response(Submit paper, Send confirmation email)
- After the paper submission, the paper will be reviewed; there can be no review without a preceding submission
Succession(Submit paper, Review paper)
- A paper can be accepted only after it has been reviewed
Precedence(Review paper, Accept paper)
- A paper cannot be both accepted and rejected
NotCoExistence(Accept paper, Reject paper)



DECLARE Templates that are Considered in the Experiment

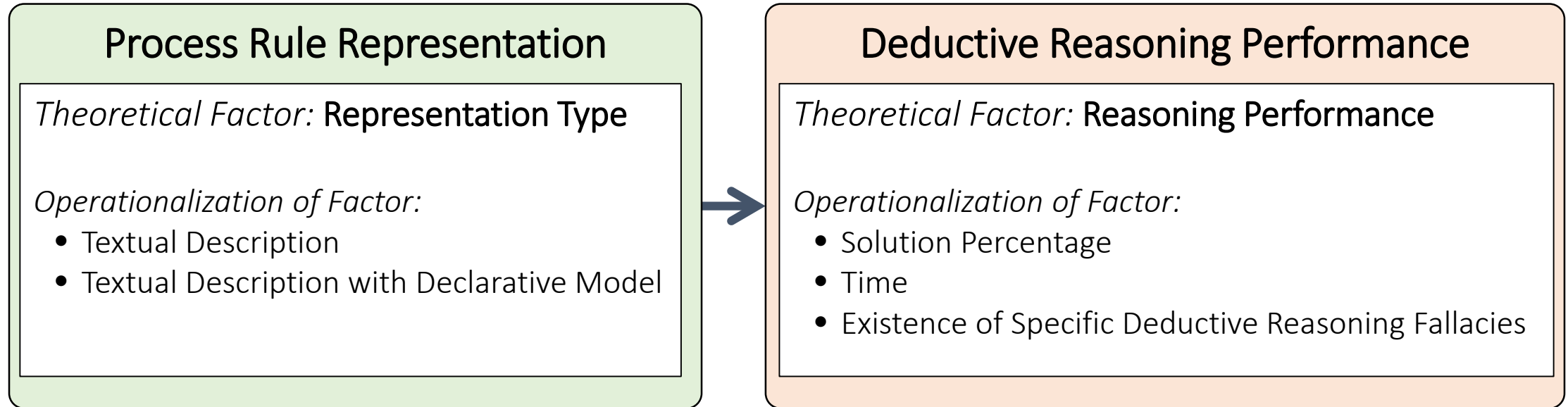
Template	Act.	Tar.	Description	Graphical notation
AtMostOne(x)		x	Activity x occurs at most once	
Participation(x)		x	Activity x occurs at least once	
Init(x)		x	Activity x always occurs first	
Last(x)		x	Activity x always occurs last	
RespondedExistence(x,y)	x	y	If x occurs, then y must occur, too	
Response(x,y)	x	y	If x occurs, then y must occur afterwards	
ChainResponse(x,y)	x	y	If x occurs, then y must occur immediately afterwards	
Precedence(x,y)	y	x	If y occurs, then x must have occurred beforehand	
AlternatePrecedence(x,y)	y	x	If y occurs, then x must have occurred beforehand, and no other y can have recurred in between	
ChainPrecedence(x,y)	y	x	If y occurs, then x must have occurred immediately beforehand	
Succession(x,y)	x, y	x, y	If x occurs, then y must occur afterwards; if y occurs, then x must have occurred beforehand	
NotCoExistence(x,y)	x, y	x, y	If x occurs, then y cannot occur; if y occurs, then x cannot occur	

Research Model

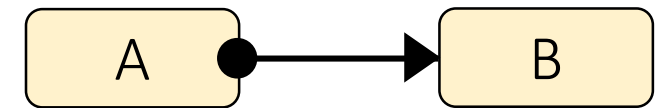


H1: Declarative process models in combination with textual representations support higher reasoning performance compared to the use of textual representations on their own.

Research Model



H2: Rules in declarative process models with directed edges that are combined with a textual representation are more likely to be misinterpreted as biconditional than rules as textual representation alone.



Humans mostly interpret directed edges as “if...and only if” (Britton and Jones, 1999) and NOT as “if...then”

Online Questionnaire: Introduction

9% filled out

Declarative Process Modeling

- Business process defined as set of constraints (rules to be respected in execution)
- No explicit specification of every possible way of execution (instead restrictions)

Next

Kathrin Figl & Claudio Di Ciccio & Hajo Reijers

Wason Selection Tasks

The Wason selection task is a famous puzzle often used in deductive reasoning research.

5 Wason selection tasks focused on single constraints were used in the experiment.

Example for the *RespondedExistence* template:

In company X the following rule has been established:

“If K occurs in the process instance, then J occurs as well.”



10. Imagine you are responsible for process compliance in this company and your task is to find out whether the rule has been violated.

The cards below represent four process instances. The process instance is already finished. On one side of each card is written whether or not K did occur, on the other side whether or not J did occur.

Your task: Please indicate all the cards that you have to turn over (i.e., all of which you need to know the information on the back) in order to find out whether the rule was violated. You can select 0-4 cards.

Not selected:

K did occur.

J did occur.

J did not occur.

K did not occur.

Selected Cards:

chosen

chosen

chosen

chosen

Wason Selection Tasks

The Wason selection task is a famous puzzle often used in deductive reasoning research.

5 Wason selection tasks focused on single constraints were used in the experiment.

Solution for the *RespondedExistence* template:

In company X the following rule has been established:

“If K occurs in the process instance, then J occurs as well.”



10. Imagine you are responsible for process compliance in this company and your task is to find out whether the rule has been violated.

The cards below represent four process instances. The process instance is already finished. On one side of each card is written whether or not K did occur, on the other side whether or not J did occur.

Your task: Please indicate all the cards that you have to turn over (i.e., all of which you need to know the information on the back) in order to find out whether the rule was violated. You can select 0-4 cards.

Not selected:

J did occur.

K did not occur.

Selected Cards:

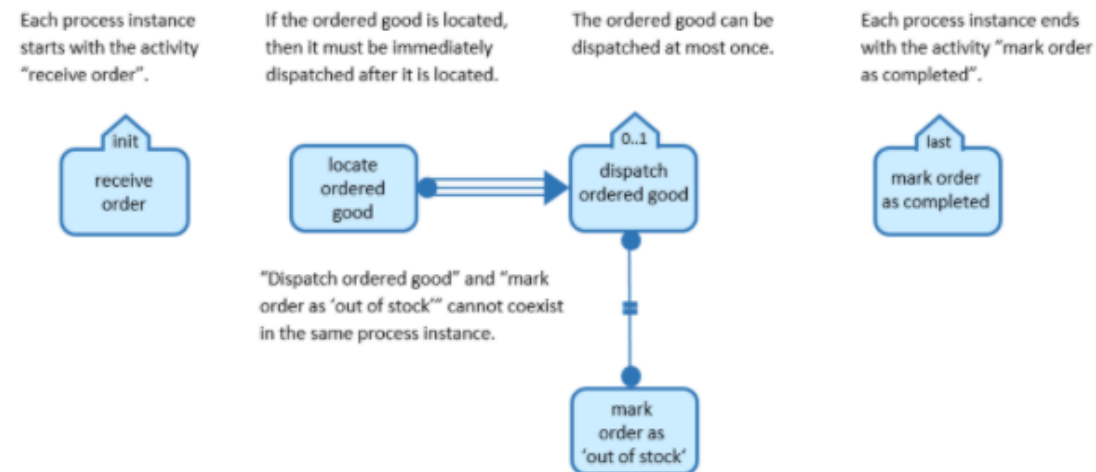
K did occur.
J did not occur.
chosen
chosen

“Order Handling” Process Model

Participants had to classify 9 process runs as “correct” or “incorrect” (or select “I don’t know”).

Although we used an online survey tool, we also provided the two models on paper to ensure readability.

Declarative Business Process Model: “Order”:



Which process sequences are valid?

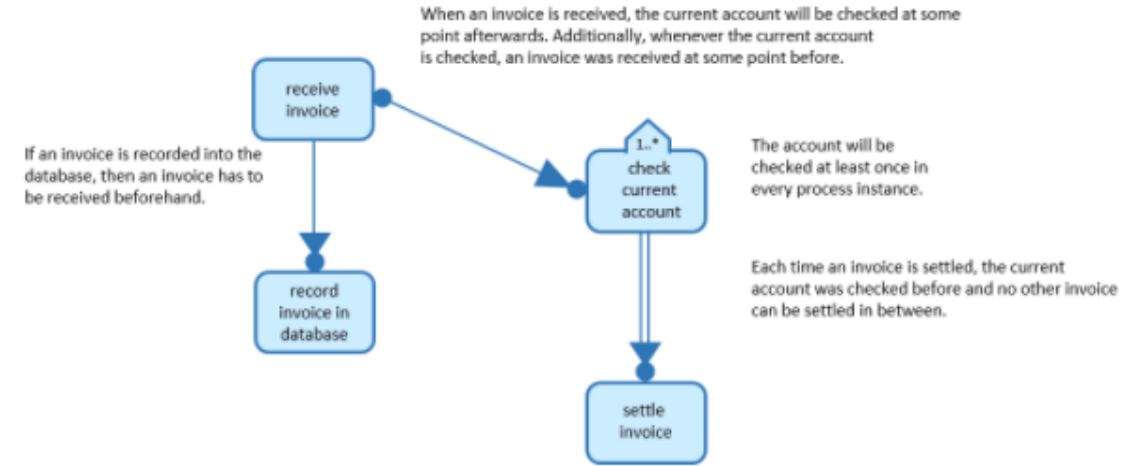
	Valid	Invalid	I don't know
“receive order”, “locate ordered goods”, “update order status”, “dispatch ordered good”, “mark order as completed”	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>
“receive order”, “locate ordered goods”, “dispatch ordered good”, “locate ordered goods”, “dispatch ordered good”, “mark order as completed”	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

“Invoice Handling” Process Model

Participants had to classify 14 process runs as “correct” or “incorrect” (or select “I don’t know”).

Although we used an online survey tool, we also provided the two models on paper to ensure readability.

Declarative Business Process Model: “Invoice”:



Which process sequences are valid?

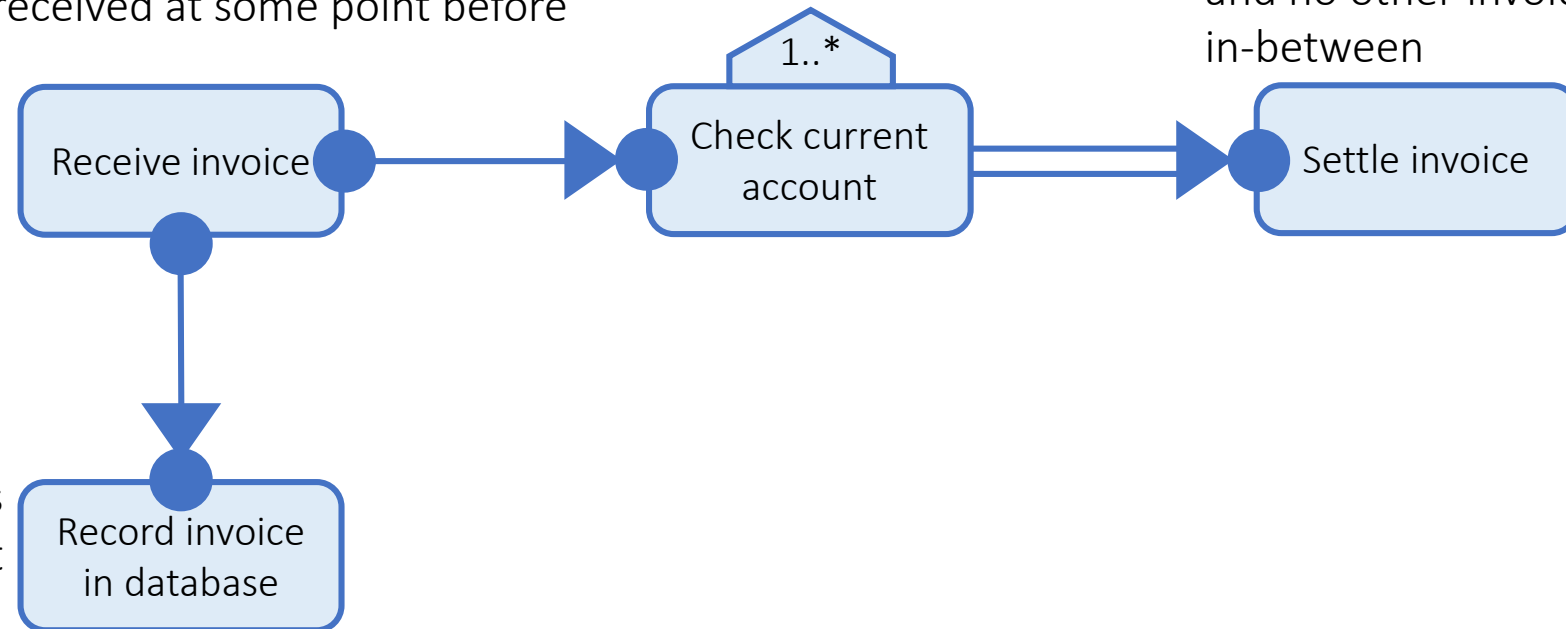
	Valid	Invalid	I don't know
“receive invoice”, “record invoice in database”, “check current account”, “settle invoice”	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>
“record invoice in database”, “receive invoice”, “check current account”	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>

“Invoice Handling” Process Model

When an invoice is received, the current account will be checked at some point afterwards. Additionally, whenever the current account is checked, an invoice was received at some point before

The account is checked at least once in every process instance

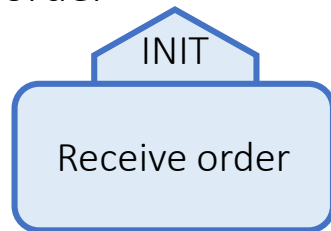
Each time an invoice is settled, the current account was checked before and no other invoice can be settled in-between



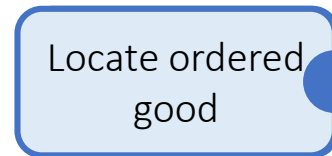
Whenever an invoice is recorded into the database, it was received beforehand

“Order Handling” Process Model

Each process instance starts with activity “Receive order”



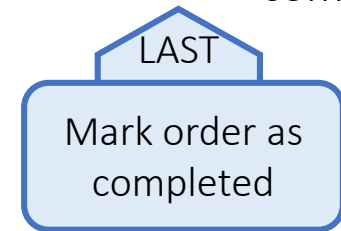
If the ordered good is located, then it must be immediately dispatched after it is located



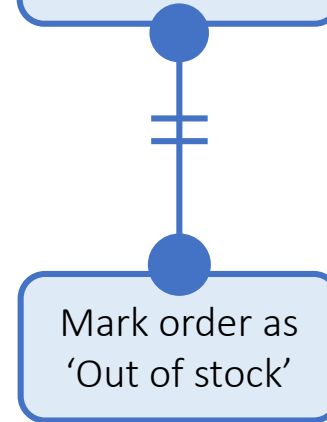
The order can be dispatched at most once



Each process instance ends with activity “Mark order as completed”



“Dispatch ordered good” and “Mark order as ‘out of stock’” cannot coexist in the same process instance



Participants

In this study, 74 information systems students from the Vienna University of Business and Economics participated voluntarily in the context of course units (in computer labs).

We chose to involve information systems students as they serve as an adequate proxy for novice corporate users of business process models.



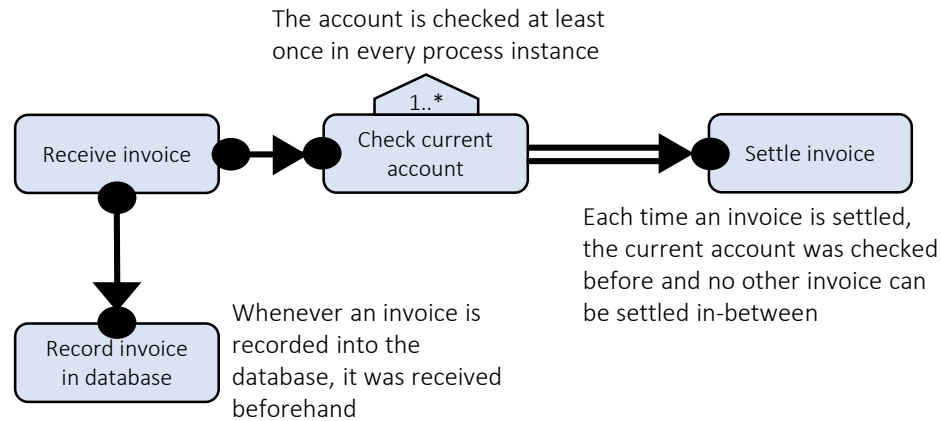
Results based on Analyses of Variance

	Text only (n=38)		Mixed text + diagram (n=37)		Stat. test
	M/count	SD/%	M/count	SD/%	
Wason selection tasks					
Solution percentage	61%	0.17	61%	0.16	n.s.
Time [sec]	59.73	25.1	77.01	76.64	n.s.
Model comprehension tasks					
Solution percentage	71%	0.17	64%	0.16	F=4.03, p=0.05
Time [sec]	212.38	86.06	212.10	79.26	n.s.
Items indicating biconditional misunderstanding					
Solution percentage	51%	0.38	43%	0.31	n.s.

Results based on Analyses of Variance

- » Time did not differ significantly between the groups in both task types.
- » No differences between experimental groups concerning solution percentages in the Wason selection tasks.
- » Significant effect of the presence of DECLARE models (in addition to the textual description) on the solution percentage in the model comprehension tasks. In contrast to the expectation behind H1, participants could answer more model comprehension tasks correctly in the text-only setting (71%) than with an additional DECLARE model (64%).
- » The DECLARE models that were part of the mixed representations did not help to prevent any of the logical errors.
- » H2 had to be rejected since the mean solution percentage of items in which biconditional misunderstanding could occur due to process model parts with directed edges was not significantly different between experimental groups.

Results based on Analyses of Variance



A more detailed analysis of the items in which the two groups differed suggests that DECLARE models were probably read as if they were procedural process models, especially if directed edges were used.

Process runs for selected reasoning tasks based on the “invoice handling” process model	Verif.	Text only (n=38)		Text + DECLARE model (n=37)		Stat. test
		Mean	SD	Mean	SD	
<“Receive invoice”, “Record invoice in database”, “Check current account”>	valid	82%	0.39	51%	0.51	t=2.88, p=0.005
<“Receive invoice”, “Record invoice in database”>	invalid	71%	0.46	49%	0.51	t=2.00, p=0.05
<“Receive invoice”, “Check current account”, “Record invoice in database”>	valid	74%	0.45	38%	0.49	t=3.30, p=0.001

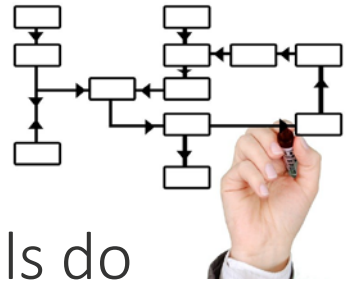
Limitations

We used artificially created snippets of declarative process models and relatively small and straightforward process models to isolate the factor of interest. External validity in the sense of generalizing the findings to more complex process scenarios will thus be limited.

Additionally, our choice of a student sample limits generalizability as, e.g., results are not generalizable to users who are already experts in using the DECLARE graphical notation. The main reason to use a student sample was to avoid an experimental bias of prior experience with declarative process modeling.



Conclusion



Overall, our preliminary findings suggest that declarative process models do not qualitatively alter human reasoning and visual process models do not outperform written language in supporting humans to understand conditional if-then arguments.

Rather, they may even confuse readers. The results gave a hint that readers of a process model tend to misinterpret declarative process models as procedural models.

Practitioners should exercise caution when tasks involve reasoning on the basis of business rules, and formal correctness of human inferences is important as logical errors might occur.

The evidence from this study further emphasizes the importance of developing understandable visual modeling approaches to business rules, to support enterprise modeling practice.

Questions and Discussion



universität
innsbruck



Prof. Dr. Kathrin Figl

kathrinfigl.com

kathrin.figl@uibk.ac.at