Figl, Kathrin, Laue, R. (2015). Influence Factors for Local Comprehensibility of Process Models. International Journal of Human-Computer Studies (Elsevier). 82, pp.96-110.

Influence Factors for Local Comprehensibility of Process Models

Suggested Running Headline: Comprehensibility of Process Models

Kathrin Figl*[a] and Ralf Laue[b]

***** (corresponding author)

[a] WU - Vienna University of Economics and Business

Institute for Information Systems & New Media

Welthandelsplatz 1, Building D2

A-1020 Vienna

kathrin.figl@wu.ac.at

Tel +43 650 979 48 89

Fax +43 1 9798383

[b] University of Applied Sciences Zwickau, Germany

ralf.laue@fh-zwickau.de

Influence Factors for Local Comprehensibility of Process Models

Abstract

The main aim of this study is to investigate human understanding of process models and to develop an improved understanding of its relevant influence factors. Aided by assumptions from cognitive psychology, this article attempts to address specific deductive reasoning difficulties based on process models. The authors developed a research model to capture the influence of two effects on the cognitive difficulty of reasoning tasks: (i) the presence of different control-flow patterns (such as conditional or parallel execution) in a process model and (ii) the interactivity of model elements. Based on solutions to 61 different reasoning tasks by 155 modelers, the results from this study indicate that the presence of certain control-flow patterns influences the cognitive difficulty of reasoning tasks. In particular, sequence is relatively easy, while loops in a model proved difficult. Modelers with higher process modeling knowledge performed better and rated subjective difficulty of loops lower than modelers with lower process modeling knowledge. The findings additionally support the prediction that interactivity between model elements is positively related to the cognitive difficulty of reasoning. Our research contributes to both academic literature on the comprehension of process models and practitioner literature focusing on cognitive difficulties when using process models.

Keywords: Deductive Reasoning, Business Process Models, Model Comprehension, Cognitive Complexity

1 Introduction

Cognitive research has got a long tradition in the context of system development. Cognitive challenges in programming – and in reading and understanding data and process models – have been studied extensively to better match system engineering methods and human cognitive capabilities (Burton-Jones et al., 2009; Gemino and Wand, 2004; Hoc et al., 1990). Unlike computers, which can easily process program code and translated conceptual models of arbitrary size, human understanding is influenced by cognitive bias and irrational beliefs (Green et al., 2009).

Process models are conceptual models commonly applied to document and communicate processes and provide between system support and organizational requirements (Rosemann, 2006). Process modeling is a critical step in the analysis and development of automated execution support for processes. Human understanding of process models is particularly relevant because process models usually involve many tasks, which "must be enacted by a human rather than a machine" (Curtis et al., 1992). However, the cognitive understanding and use of such models may be error-prone, especially for novices. Therefore, human interaction with process models is a relevant new research field. Several attempts have been made to identify influence factors of process model understanding (e.g., Figl et al., 2013a; Figl et al., 2013b; Mendling et al., 2012; Reijers and Mendling, 2011) and process model creation (e.g., Recker et al., 2012).

In this article, we focus on how humans reason on the basis of process models. While a variety of previous studies in this research stream have related model comprehension to global complexity metrics of process models (e.g., size, the number of specific model elements, labeling, layout,…) (Mendling et al., 2010b; Mendling et al., 2012; Reijers and Mendling, 2011), little is known about *what exactly* makes it difficult for humans to reason on the basis of a process model. It is in particular the comprehensibility of *local* properties of model structures as well as the interactivity between model elements that have not been studied in detail. Therefore, this article examines the cognitive difficulty of understanding specific parts of a process model instead of considering the model as a whole. Theoretically, it builds on cognitive load theory to explain cognitive difficulty of reasoning tasks. We propose to conceptualize comprehension of process models as deductive reasoning tasks, with the process model as the premise, and the comprehension tasks as possible conclusions drawn on the basis of the model.The article builds on a data set of comprehension questions that allows us to evaluate the cognitive difficulty of reasoning tasks and to relate this value to local metrics of the model elements involved in the task.

Related research efforts have already been undertaken in the area of software complexity (e.g. Yang et al., 2005). In that context, researchers have, for instance, identified measures that assign complexity values to portions of the code. By visualizing such measures in combination with code lines, the reader of a program could be alerted that a specific part of the code required special attention (Umphress et al., 2006). Likewise, knowledge gained in this study could inform modeling tool designers about process model structures with a sophisticated cognitive difficulty which enables them to design similar tool-based feedback. From a theoretical perspective, this study makes a contribution to the body of literature by providing the first empirical analysis of relevant influence factors for local comprehensibility of process models.

1 Deductive Reasoning with Process Models

1.1 Deductive Reasoning

Both comprehension and correct interpretation of models are relevant for many different tasks (Burton-Jones et al., 2009). In this context, Dumas et al. (2013) state that “a thorough understanding is the prerequisite to conduct process analysis, redesign or execution.” Asking comprehension questions is therefore the most common way to measure comprehension of process models (e.g. Mendling et al., 2012; Reijers and Mendling, 2011). Such comprehension questions can be characterized as deductive reasoning tasks, since correct answers can be derived from general knowledge on process-flow logic and the specific process model. The questions require deductive reasoning, which is defined as the “mental process of making inferences that are logical” (Johnson-Laird, 2010). While the “classical” psychological research on deductive reasoning has predominantly focused on propositional (based on negation and connectives as *if*, *or* and *and*) and predicate reasoning (based on quantifiers as *all, some* or *no*), concepts related to process logic have largely been neglected.

In deductive reasoning, a clear distinction is made between content and form. For instance, in the case of the form modus ponens with two premises (A implies B, A is true), the conclusion (B is true) is always valid if the premises are true, regardless of the premises' content. A and B can be substituted by any content and the conclusion will still be valid. For process models this means that the verbal labels in the models and comprehension tasks could be substituted by any kind of label, e.g. abstract numbers, and the logical soundness of a conclusion would still be the same. Figure 1 provides an example of a process model with abstract labels and four sound conclusions regarding the two model elements D and H. The conclusions refer to a single process instance, i.e. a single execution of a business case according to the rules described in the business process model. Process instances are created and executed based on the process logic defined in the model (Rinderle et al., 2004). The model uses the widespread Business Process Model and Notation (BPMN) standard. Rectangles with rounded corners depict a task. Arrows between elements indicate in which order the tasks can be executed. The diamond symbol is used to model a decision (thus, in a single process instance either task H or task I is executed, but not both), and the diamond symbol with a "+" symbol inside is used to model the start and end of parallel execution.

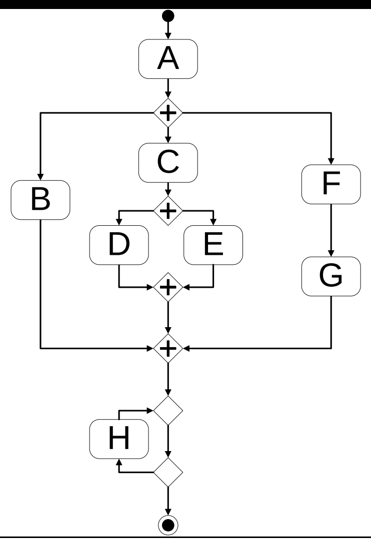
Premises: An example process model	Possible sound conclusions for a single execution of the process
	**D** and **H** cannot be executed in parallel. **H** can be executed more often than **D.** **D** and **H** can both be executed in a process instance. **D** is executed before **H.**

Figure 1. Process model comprehension tasks as reasoning tasks.

A typical outcome in research on deductive reasoning research often is the use of frequency tables of the correct solutions to different logical arguments to better analyze how humans intuitively reason and to contrast their reasoning with formal logic (e.g. Beller and Spada, 2003; Braine et al., 1995). A major result of such studies is that humans do not necessarily reason logically but apply heuristics and are often subject to fallacies. For instance, according to the "post hoc ergo propter hoc" fallacy, humans assume "that a particular event, B, is caused by another event, A, simply because B follows A in time"

(Damer, 2013). Thus, humans tend to misinterpret a temporal sequence for a causal connection. By the same token, we are interested in how far humans reason logically on the basis of process models, whether specific reasoning fallacies do occur and whether some inferences are more difficult than others. In the following sections, we want to discuss several influence factors for the cognitive difficulty to reason on the basis of a process model.

1.2 Cognitive Load and Deductive Reasoning

From a cognitive point of view, the human working memory is the main component involved in deductive reasoning with process models. The term ‘working memory’ “refers to a brain system that provides temporary storage and manipulation of the information necessary for such complex cognitive tasks as language comprehension, learning, and reasoning” (Baddeley, 1992). If working memory is overburdened, reasoning errors are more likely to occur (De Neys et al., 2005; Süß et al., 2002).

In contrast to typical deductive arguments (in the form of two premises and a conclusion), process models as premises are not single but compound premises which makes deductive reasoning tasks fairly complex. So far, no current theory has explicitly addressed cognitive load demands in reasoning with process models. However, we can draw on theories from related areas, e.g. profound theories on the cognitive processes that are performed by programmers to understand a piece of software. The challenge to reason on the basis of a process model is fairly similar to the process of understanding facts from software code: (i) Control-flow structures such as conditional execution or loops need to be considered; (ii) Control-flow structures can be nested, and the information (the process model or the code) can be traced by the reader in an arbitrary order. Therefore, it is reasonable to assume that the process of reasoning in a business process model can be described as an adaption of the model for the process of program understanding described by Cant et al. (1995). This model suggests that, in order to understand a section of code, the programmer performs both chunking and tracing. Chunking refers to recognizing a group of statements and memorizing it as a single reference (a “chunk”). In business process models, chunking involves identification of model elements (such as tasks and gateways) that can be considered a group of elements “belonging together”. Control-flow patterns (van der Aalst et al., 2003) are typical structures for building such groups of related elements. Tracing can be described as scanning through a program (or a business process model) in order to identify relevant chunks. Cant et al. (1995) point out that the cognitive difficulty of solving a programming inquiry is determined by the complexity of the chunks and the difficulty in moving between them. Thus, when investigating reasoning in process models, we also need to consider the following two aspects: (i) the types of involved *control-flow patterns* (as candidates for chunks) and (ii) the relations between the elements representing those patterns, affecting moving between them (*element interactivity*). Bearing this in mind, the following sections will detail the cognitive load of element interactivity and control-flow patterns in process models. We will then describe how human process modeling expertise may ease identification of patterns in process models and reduce cognitive load. Finally, we also discuss whether the phrasing of a reasoning task may increase potential reasoning bias.

1.2.1 Element Interactivity

Process models usually include more elements than those needed to solve a specific reasoning task. Not all elements of a process model are therefore of equal importance to a specific task. On the contrary, it might be sufficient to understand just a small detail of the model to find a correct solution. Thus, when reasoning on various pairs of elements in a model, the reasoning difficulty will vary depending on the selection of those elements. The level of interactivity between the elements determines the number of model elements that really require attention. Elements interact if interrelated, such, that it is necessary to assimilate them simultaneously (Sweller, 1994). High interactivity results in high cognitive load, because each element has to be processed with reference to other elements. In contrast, cognitive load is low when the elements can be processed serially, without referring to other elements. Indeed, empirical studies have revealed that a high interactivity between model elements can make a model more difficult to understand (Guceglioglu and Demirors, 2005; Vanderfeesten et al., 2008). This line of reasoning is further supported by the fact that with a higher number of model elements, overall cognitive difficulty and the number of errors in the models increase (Mendling et al., 2007a; Mendling et al., 2010a).

1.2.2 Control-Flow Patterns

Second, we turn to control-flow patterns (control structures) in process models and the cognitive difficulty to understand them. Control-flow patterns refer to “activities and their execution ordering through different constructors” (van der Aalst et al., 2003). Execution ordering involves basic control structures such as *Sequence*, *Loops* (also known as *iteration*, or *cycles*), the *AND*-pattern (parallel execution) and the *XOR*-pattern (conditional execution). For the purpose of this article, we examine the impact of the most basic control-flow patterns as summarized in Table 1.

Table 1. Description of control flow patterns.

Control-flow pattern	Description
Sequence	Tasks are executed in succession.
AND	A pair of an AND-split and an AND-join, allowing ≥ 2 paths to be executed in parallel (control-flow patterns “Parallel Split” and “Synchronization”)
XOR	A pair of an XOR-split and an XOR-join with the meaning that exactly one out of ≥ 2 possible paths is chosen and executed (control-flow patterns “Exclusive Choice” and “Simple Merge”).
Loop	A loop in the model that allows the repeated execution of some part of the model (can be one of the control-flow patterns “Structured Loop” and “Arbitrary Cycles”).

We are interested in whether the control-flow patterns *AND*, *XOR* or *Loop* are more difficult to understand than the simple *Sequence* pattern and whether, as a consequence, cognitive difficulty of deductive reasoning tasks differs depending on the control-flow patterns involved. So far, few studies have been performed on the cognitive aspects of understanding such control structures in process models. However, based on the similarity between structures in software code and process models (Guceglioglu and Demirors, 2005; Vanderfeesten et al., 2008), we can draw on research findings on program code complexity as a basis for hypothesizing about the cognitive difficulty of control-flow patterns. The area of procedural code complexity (Tegarden et al., 1995) considers sequences, decision structures and loops and, therefore, is an appropriate equivalent to analyzing control structures of process models. Cater et al. (1984), for instance, propose to calculate logical effort of code on the basis of decomposing a program into structural elements like loops or decisions. Cant and Jeffery (1995) argue that “intuitively, a conditional control structure is more complex to understand than a normal sequential section of code.” Shao & Wang (2003) propose different cognitive weights for basic control structures. They rate sequence as the easiest (weight=1), followed by branching (XOR) (weight=2), iteration (loops) (weight=3), embedded components (weight=2-3) and parallel execution (AND) as most difficult (weight=4). However, a serious weakness of this proposal is that it is not based on empirical evidence.

Previous research on process models suggests that understanding their control-flow is generally difficult for humans (Mendling et al., 2010a). Current research results revealed that some kinds of control-flow constructs are more difficult to understand than others. However, different authors provide seemingly conflicting positions and no single study exists which adequately covers comprehension of control-flow constructs. For example, Sánchez-González et al. (2012) conclude that XORs are more difficult to understand than AND patterns. In contrast, Weitlaner et al. (2013), who investigated the cognitive difficulty of control-flow elements in a comprehension study with practitioners, found slightly lower comprehension scores for concurrency in comparison to order, XOR and repetition. Based on a small number of user feedbacks, they reported specific problems of users with concurrency. However, it remains unclear whether the differences they had found were statistically significant and whether they actually resulted from problems understanding *AND* or rather from the fact that the practitioners simply did not know the notation. Modeling guidelines recommend to avoid inclusive OR gateways altogether, as they may lead to reasoning fallacies (Mendling et al., 2010a). The higher cognitive difficulty of inclusive ORs is also reflected in findings on deductive reasoning with natural language connectives. “Or” is more likely to be interpreted in its exclusive form, not as an inclusive “or”-operator (Naess, 1961). Based on the low relevance of inclusive ORs in modeling practice, we refrain from including them in our study.

1.3 Modeling Knowledge

In empirical research on system development, a growing body of literature has investigated how novices and experts differ (e.g. Davies, 1994; Gilmore, 1990). For instance, when less experienced modelers create new conceptual models, they demonstrate more difficulties in understanding the problem and integrating problem facets than modelers with profound modeling knowledge do. This leads to lower quality models regarding several characteristics such as correctness, completeness or innovativeness (Batra and Davis, 1992; Shanks, 1997). Thus, modelers with higher modeling knowledge are not only faster but their cognitive processing also changes in a qualitative way. Similarly, in the context of conceptual modeling, Petre (1995) has claimed that “experts ‘see’ differently and use different strategies than novice graphical programmers”. Studies have revealed, for instance, that they develop language-independent, abstract problem representations in their mind, e.g., for iterations (e.g. Rist, 1989). These ‘schemas’ – constructs that merge multiple elements of information into one concept – are stored in long-term memory. This way, the working memory does not have to store those elements individually but can deal with a ‘schema’ as just one piece of information. Consequently, there are more free working memory resources for the same reasoning task. Unlike novices, people with a higher programming knowledge memorize program structures as patterns. When developing program elements, they plan on a higher, abstract level (Bateson et al., 1987).

1.4 Validity of Conclusion

While it can be objectively determined whether a given conclusion is valid in a deductive reasoning task, its phrasing as well as the validity of the conclusion may lead to reasoning biases. According to the “atmosphere-effect” (Woodworth and Sells, 1935) and the “matching-strategy” (Wetherick and Gilhooly, 1995), wording of the premises and conclusion is relevant for the relative difficulty of deductive reasoning tasks. For instance, if the premises are affirmative, the participants are more likely to accept an affirmative conclusion (Woodworth and Sells, 1935). To give an example, based on the two premises “If some X’s are Y’s, and some Y’s are Z’s”, studies show that 72% of humans tend to wrongly accept the invalid conclusion “then: Some X’s are Z’s” (Wetherick and Gilhooly, 1995; Woodworth and Sells, 1935). Thus, they employed a matching strategy instead of relying on deductive reasoning.

2 Research Model

Based on the theoretical assumptions described above, we will now discuss the anticipated effects of the four supposed relevant influence factors on the cognitive difficulty of a deductive reasoning task (viz., a specific comprehension task based on a process model). We summarize our expectations in the research model shown in Figure 2. Overall, our model aims to contribute to research on reading, analyzing and using process models. The ability to deduce correct conclusions based on the ‘premises’ expressed in a process model is relevant to almost any situation in which the comprehension of the model is necessary for analysis or reengineering tasks (Burton-Jones et al., 2009).

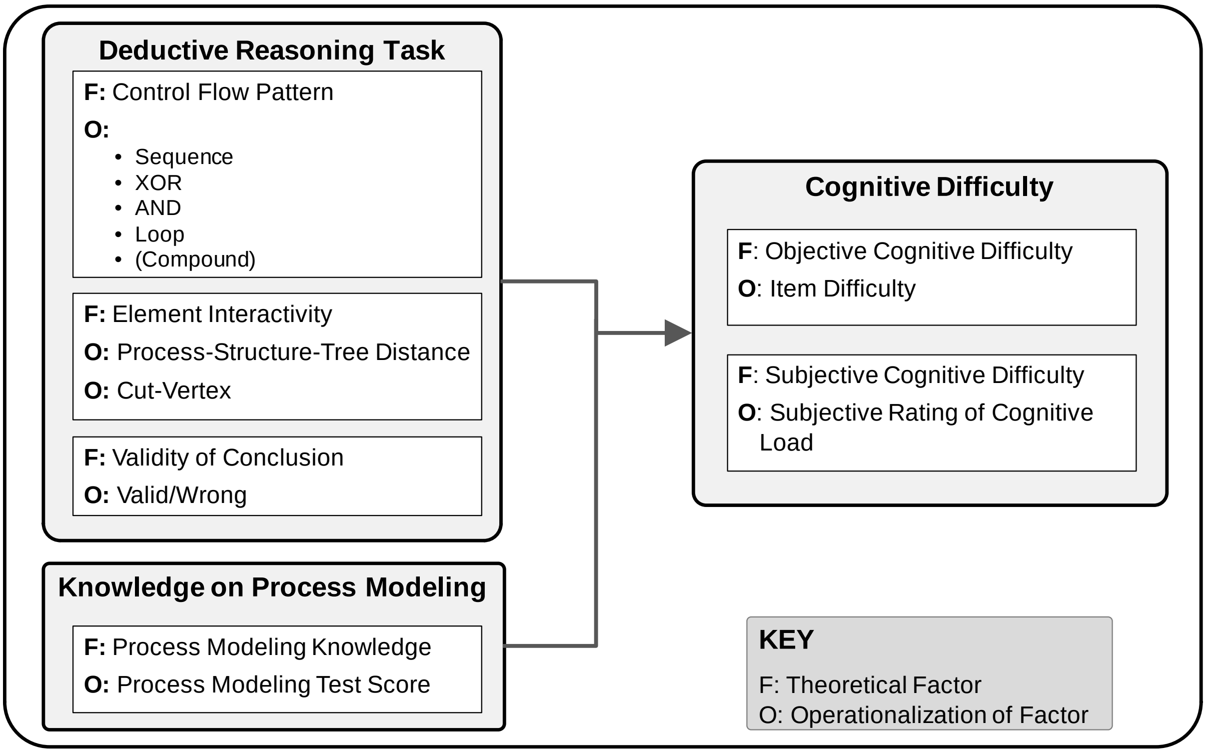


Figure 2. Research model.

First, we turn to the influence of expertise in the area of process modeling. In light of the theoretical considerations on novice and advance modelers above, it can be assumed that there are differences between modelers with lower or with higher process modeling knowledge when it comes to logic reasoning on a process model: While the modelers with higher process modeling knowledge presumably have stored schemas in long-term memory that allow them to process a group of model elements (such as all those elements that build a certain control-flow pattern) simultaneously, modelers with lower process modeling knowledge have to split their cognitive resources for the individual model elements. The higher the cognitive load the lower the ability to make valid conclusions. This is in line with the results by Mendling et al. (2012) who have shown a positive influence of modeling knowledge and experience on the ability to understand process models. Thus, we contend that modelers with higher process modeling knowledge will be better in solving deductive reasoning tasks and will also experience these tasks as easier than modelers with lower process modeling knowledge. Therefore we assume:

H1. It is more difficult for modelers with lower process modeling knowledge to solve deductive reasoning tasks in a process model than it is for modelers with higher process modeling knowledge.

Second, we turn to different control-flow patterns. Previous research comparing different control structures in program code (Cant et al., 1995; Shao and Wang, 2003) and in process models (Sánchez-

González et al.) suggests that cognitive difficulty of comprehending control-flow patterns varies. Therefore, we contend that the cognitive difficulty of a reasoning task is influenced by the type of involved control-flow patterns. Therefore we assume:

H2. The type of control-flow patterns that has to be understood (such as *Sequence*, “parallel split” [*AND*] for parallel execution or “exclusive choice” [*XOR*] for conditional execution and *Loop*) has an influence on the cognitive difficulty in reasoning.

While there are compelling arguments from literature that *Sequence*, as the very basic control-flow pattern, has lower cognitive difficulty than other more complex patterns (Cant et al., 1995), previous research does not suggest further clear-cut differences among the cognitive difficulties of different control-flow patterns. However, we assume that a reasoning task will be more difficult when it involves a combination of multiple patterns that would probably add up to a greater cognitive load. We will call a combination of more than one pattern other than the simple *Sequence* pattern (viz., a combination of parallel or conditional execution and iteration) a *Compound* pattern. Please note that this term does not refer to a standardized pattern, it is rather used as an abbreviation for “more than one control-flow pattern other than *Sequence*.” We hypothesize that such *Compound* patterns are of a higher cognitive difficulty than single patterns. Therefore, we are particularly interested in the following hypotheses:

H2a. Reasoning tasks that require only the *Sequence* pattern to be understood are easier to solve than those for which other (non-trivial) patterns are involved.

H2b. Reasoning tasks that require a *Compound* pattern (i.e., more than one control-flow pattern [other than the simple *Sequence* pattern]) to be understood are more difficult to solve than reasoning tasks for which only a single control-flow pattern has to be considered for finding the correct answer.

Third, we turn to the interactivity between elements. We expect that if it is necessary for a reasoning task to consider a high number of model elements and their interrelations, the cognitive load will be higher (Sweller, 1994). On the basis of this argument, we thus propose the following hypothesis:

H3. The interactivity between elements will be positively associated with the cognitive difficulty of a reasoning task including them.

Fourth, we turn to the validity of the conclusion. According to the atmosphere and the matching hypotheses, it is relevant whether there is a consistency between affirmativeness of premises and conclusions. We speculate that a typical process model is an “affirmative” presentation of the premises, in case the conclusion is valid. This is because typical process models are imperative, meaning that they “require all execution alternatives to be explicitly specified in the model” (Pichler et al., 2012). Unlike the uncommon “declarative” process models, which would focus on constraints and impossible process executions, an imperative process model specifies all possible alternatives and visually presents all possible instantiations of the process. If an (affirmatively formulated) conclusion is valid, the “affirmative” nature of the process model represents the same affirmative “atmosphere”, and, as a result, the “atmosphere” bias supports providing a correct answer. Thus, we anticipate the following:

H4. It is easier to correctly identify (affirmatively worded) valid conclusions of a process model than invalid conclusions.

3 Design and Measures

In order to test our hypotheses, we use a subset of a large data set of answers to process model comprehension questions *(Figl et al., 2013b)*.[1] We want to point out that we analyzed comprehension values on task level (viz., each comprehension question constitutes a specific reasoning task). Answers to the questions were aggregated (across all participants) for two groups: modelers with lower modeling knowledge and modelers with higher modeling knowledge. Thus, we obtained four averaged estimates of cognitive difficulty for each reasoning task (one for subjective and one for objective difficulty, for each of the two groups, respectively). In the following section, we present the background of the dataset, the construction of the process models and the selection of measures used to determine subjective and objective cognitive difficulty of reasoning tasks. Then, we describe how we measured element interactivity and type of control flow pattern relevant to a reasoning task.

[1] This larger data set has been used to study another research question – the understandability of different visual gateway symbols. By selecting the data subset for which the design of the symbols did not impose additional burdens we avoided unnecessary "noise" and variance in answers.

3.1 Materials: Questionnaire Parts

The questionnaire was prepared for a pencil- and paper evaluation. The first part comprised items on subjects' demographic data, academic qualifications and modeling experience. We asked the subjects to rate the amount of process models they had already created or read and to estimate the amount of hours of modeling training attended (at school or at university).

The next part of the questionnaire contained questions adapted from the theoretical knowledge test of process modeling by Mendling et al. (Mendling and Strembeck, 2008; Mendling et al., 2012) . Examples of test items were “Exclusive choices can be used to model repetitions” and “If an activity is modeled to be part of a loop, it has to be executed at least once”. Using a previous test, we follow a suggestion of Siegmund and Schumann (2014) on how to measure experience in the context of comprehension experiments: “researchers can use a validated instrument… instead of using an ad hoc definition that differs between different experiments and researcher groups.” This way, the subjects’ subjective ratings of their experience with process models were complemented with an objective measurement of process modeling knowledge. Experience is a major confounding parameter in such comprehension experiments.

Moreover, the questionnaire included a tutorial on process modeling, which was intended to recall the meaning of each symbol used and covered all aspects the subjects would need to know in order to perform the reasoning tasks. The main part of the questionnaire contained four different process models with eight corresponding deductive reasoning tasks per model. The following two sections 3.2 and 3.3 describe details of the choice of models and tasks. Appendix A gives an example of a process model and its corresponding reasoning tasks.

To measure the perceived (subjective) difficulty of the reasoning tasks, we asked the subjects to rate each task on a seven-point, single-item cognitive load measure (with the labels “very difficult”, “difficult”, “rather difficult”, “neither difficult nor easy”, “rather easy”, “easy” and “very easy”) as proposed by Marcus et al. (1996). To avoid order effects due to decreasing motivation or concentration of subjects we used two different scramblings. Models as well as reasoning tasks were presented in different orders. The subjects were allowed to spend as much or as little time as desired on the questionnaire.

3.2 Choice of Process Models

We used four different models as a basis for presenting the deductive reasoning tasks. Two of them were drawn from the business domain (Product Management and Customer Relationship Management). The other two were taken from relatively uncommon domains: an emergency process plan for drinking-water pollution and an e-mail election process (the last process was taken from the BPMN standard document, Object Management Group, 2013).

We used models with concrete instead of abstract labels in order to make results valid to real models in practice. We chose to use a verb-object labeling style as this is the most frequently suggested naming convention (Leopold et al., 2013). Abstract labels might limit the generalizability because research revealed that material with abstract labels has a slightly different effect on human reasoning than material with content labels (Beller and Spada, 2003; Markovits et al., 2002), which also eases model comprehension (Mendling et al., 2012).

3.3 Choice of Reasoning Tasks

Eight different reasoning tasks were devised for each of the four models. This is in concurrence with Evans (1972) who proposed to use a variety of reasoning tasks to be able to relate results to underlying cognitive operations. Too few reasoning tasks could result in the misinterpretation of single sentences in the tasks by the subjects and invalid results. The wording of the tasks was based on comprehension questions developed by Reijers et al. (2011) and Melcher et al. (2010) and refers to different relations between model elements. For each model, we asked two questions for each of the following question types: “A and B can be executed at the same point of time”, “A and B can be executed in parallel”, “In one process instance A as well as B can be executed”, “The process steps A and B are mutually exclusive”, “A can be executed more often than B”, “In each process instance A is executed exactly as often as B”, “If A as well as B are executed in a process instance, then A is executed before B” and “If A as well as B are executed in a process instance, then A has to be finalized before B can start.” Unlike Reijers et al. (2011) and Melcher et al. (2010), we used consistent questions so that subjects had always to consider two model elements (two activities). This was necessary for the comparison of the results of the reasoning tasks with each other. Although the individual process element labels used in the reasoning tasks were meaningful (e.g. “dig off soil” and “buy new equipment”), we assured that the correctness of the stated relation between them (e.g. “are mutually exclusive”) could only be answered based on the process model and not based on every-day knowledge (as for instance in a reasoning task such as “‘Accepting offer’ and ‘declining offer’ are mutually exclusive”).

All comprehension questions were formulated in a way that they did not include negations (e.g., we ask whether “A and B *can* be executed in parallel”, and not “A and B *cannot* be executed in parallel”.) Additionally, we ran a pre-test to ascertain that the wording was comprehensible for the subjects (Laue and Gadatsch, 2011).

When compiling the test material, we alternated correct and incorrect answers. This variation was required to investigate the factor “validity of conclusion”. To further enhance the variety of different reasoning tasks, we used two versions of the questionnaire (version A and B), resulting in 64 different reasoning tasks. In both versions, exactly the same models were used.

3.4 Measurement of Element Interactivity

In this section, we will discuss how we operationalized element interactivity in the process models through two different metrics. For defining these metrics, we consider a process model as a directed graph without referring to the semantic meaning of its nodes. In addition, the measures do not take into account which types of gateways are relevant for answering a question, since the factor “control-flow patterns” covers this point.

3.4.1 Process Structure Tree Distance

In order to define a measure for the cognitive load resulting from a reasoning task involving two elements in a process model, we follow the idea of Vanhatalo et al. (2009) to decompose the process model into canonical fragments with a single entry and a single exit. These fragments can be arranged in a process-structure tree in a way that there is exactly one process structure tree for each process model. Figure 3 shows an example process model and its corresponding process tree.

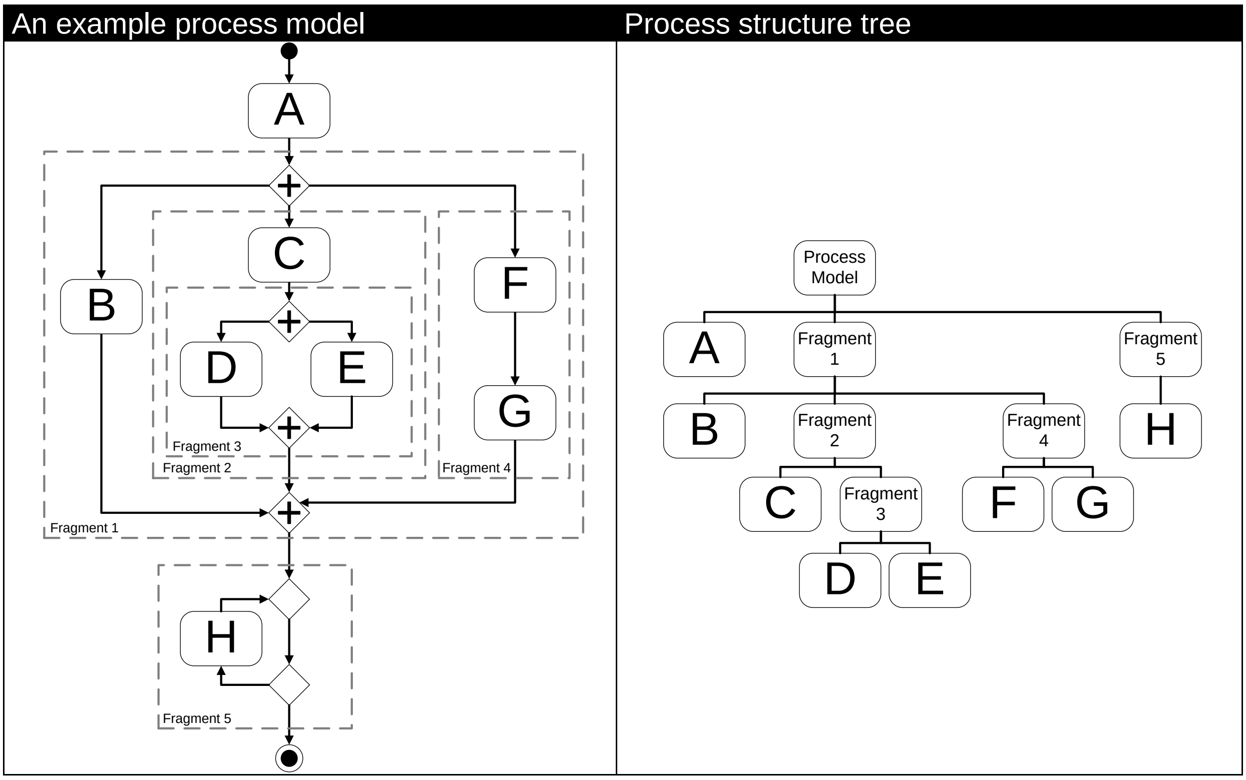


Figure 3. Example process model and its corresponding process tree.

We argue that the distance between two elements in the process structure tree can serve as a measure for the interactivity between those elements. Each fragment in the tree represents one concept (for example, the concept of an exclusive choice or the concept of parallel branching) that the reader of the model has to understand. If elements are located in deeply nested control-flow blocks, the reader has to understand a large number of concepts before being able to answer a question concerning the relation between those elements. On the other hand, if both elements are located in the same control block without additional nesting, they will also be in the same region of the process structure tree.

Formally, we define the *process structure tree distance* between two elements X and Y of a process model as the number of edges between X and Y in the process structure tree, minus one. This means that elements in a sequence or in the same control block (for example, two elements that are executed in parallel without any further branching) have a process structure tree distance of one. For instance, the process structure tree distance between model elements D and H in Figure 3 is 5, while the distance between model elements D and E is 1.

3.4.2 Cut-Vertices

A second aspect we took into account when discussing the interactivity between elements A and B in a process model is the case where a single edge in the process model separates the model into two disjoint parts P_1 and P_2, such that $A \in P_1$ and $B \in P_2$. In terms of graph theory, this means that the connected graph, that forms the process model, has a "cut-vertex" on a path from A to B, i.e., a vertex (edge) that, when removed, causes a disconnection in the remaining graph. If such a cut-vertex exists between A and B, the mental model of the relationships between A and B becomes much easier, because A is located "before" and B is located "after" an easy-to-spot reference point (the cut-vertex). The model in Figure 3 shows, for instance, two cut vertices (between model element A and fragment 1, and between fragment 1 and fragment 4).

3.5 Measurement of Control-Flow Patterns

We used a consensus-building rating approach to determine which control-flow patterns had to be considered to solve each deductive reasoning task. First, two raters (the authors of the paper) made the judgment independently, and in a next step, inconsistencies were discussed to reach a final categorization. All 64 reasoning tasks were categorized to refer to one of the control-flow patterns *Sequence*, *AND*, *XOR*, *Loop* and *Compound*. The *Compound* category was used for reasoning tasks which demand participants to understand more than one control-flow pattern other than *Sequence*.

3.6 Subjects

A total of 199 business students participated in this study (125 males, 74 females). We categorized subjects into two groups according to their score in the process modeling knowledge test. We selected this test instrument to divide the students into a group with higher process modeling knowledge and another group with lower process modeling knowledge, because previous research has shown that theoretical knowledge is more important to syntactical process model comprehension than other factors such as practical experience (Mendling et al., 2012). The use of groups is important for our subsequent analysis, as we intend to perform statistical tests on the basis of reasoning tasks as test subjects and not on the basis of subjects, which would allow the use of the process model knowledge test score as continuous covariate. Based on a median-split of the process modeling test scores (the median was 5 correct answers), we grouped subjects in two equally large (extreme) groups: 72 subjects (36%) with 0-4 points (0-50% test score) and 84 subjects (42%) with 6-8 points (75%-100%). We justify this selection with the fact that according to Preacher et al. (2005) extreme groups “need not be equal in size or cover the same range of scores”. The remaining 44 subjects (22%) with five correct answers were excluded, resulting in a sample size of 156; 72 subjects for the group with lower process modeling knowledge and 84 subjects for the group with higher process modeling knowledge. Table 2 provides a summary of the subject groups and their characteristics. As expected, the subjects of the higher process modeling knowledge group had created and read significantly more process models and had been trained for more hours in modeling than subjects of the other group (see Table 2).

Table 2. Sample description and differences between modelers with lower / higher process modeling knowledge

	Low Process Modeling Knowledge (n=72)		High Process Modeling Knowledge (n=84)		Total (n=156)		*Statistical Test*
	Mean/ Number/%	SD/%	Mean/ Number/%	SD/%	Mean/ number	SD/ %	
Gender							
Female	38	53%	17	20%	55	35%	-
Male	34	47%	67	80%	101	65%	
Age	22.58	3.18	24.74	4.01	23.74	3.79	-
Highest degree completed							
High school	31	43%	19	23%	50	32%	
One or more years of university	36	50%	48	57%	84	54%	-
Bachelor	4	6%	10	12%	14	9%	
Master	1	1%	7	8%	8	5%	
Process modeling test score	38%	0.13	81%	0.08	61%	0.24	*$T_{df=153}$=-25.03 p=0.000*
Amount of process models created or read	4.08	11.02	29.70	49.18	17.88	38.92	*$T_{df=154}$=-4.33, p=0.000*
Amount of modeling training (in hours)	6.96	19.83	26.46	30.36	17.51	27.74	*$T_{df=144}$=-4.50, p=0.000*

4 Data Preparation

In a first step, we analyzed the quality of the reasoning tasks from a test-theoretical point of view. For this purpose, we calculated two indicators: the discrimination coefficient (the correlation between single item and total score of a subject) and discrimination index (the difference between the extreme groups of the 27% best and worst subjects, based on the total score, cf. Matlock-Hetzel, 1997). As cut-off value, we used the critical value of 0.17 (that results from considering a significant Pearson correlation on a

5% significance level for a one-tailed test and a sample size of 90) for the discrimination coefficient and 0 for the discrimination index. A negative discrimination index or a non-significant correlation between a single item and the total score indicates that generally low-performing subjects scored better on a task than high-performing ones. Such a result suggests that the wording of items might have been unclear or ambiguous. On the basis of this analysis, we excluded three of the 64 comprehension questions that did not meet the defined criteria.

5 Results

In this section, we report on the results relating to our hypotheses. For this purpose, we performed two multivariate analyses of covariance (MANCOVA) tests, using SPSS 11. The objective and subjective cognitive difficulties of the modelers with lower and higher modeling knowledge were used as dependent variables so that process modeling knowledge (lower vs. higher) was a within-subject factor for each MANCOVA. We included three independent variables in each of the two analyses (for the dependent variables – objective and subjective cognitive difficulty): (i) control-flow patterns with five levels (*Sequence*, *AND*, *XOR*, *Loop*, *Compound*), (ii) validity of conclusion with two levels (valid, wrong), (iii) existence of a cut vertex (existent, nonexistent) and one covariate – the process structure tree distance. Table 3 provides all results (F and η^2 are omitted in case of non-significant results with $p>0.11$).

Table 3. Experimental results: Influence of deductive reasoning tasks on cognitive complexity. [Please note that the term "subject" in the table refers to reasoning tasks, not to the participants who had answered them.]

Dependent Variable	Factor		F ($df_{Hypothesis}$, df_{Error})	p	η²
Objective Cognitive Difficulty (%)	Within-Subject Effect	Process modeling knowledge	$8.05_{1,53}$	**0.006**	0.13
	Between-Subject Effect	Control-flow pattern	$3.19_{1,53}$	**0.02**	0.19
		Element Interactivity: process structure tree distance	$22.08_{1,53}$	**0.000**	0.29
		Element interactivity: cut-vertex		>0.11	
		Validity of conclusion	$7.53_{1,53}$	**0.008**	0.12
	Interaction Effect	Process modeling knowledge * control-flow pattern	$2.71_{1,53}$	0.04	0.17
		Process modeling knowledge * process structure tree distance		>0.11	
		Process modeling knowledge * cut-vertex		>0.11	
		Process modeling knowledge * validity of conclusion		>0.11	
Subjective Cognitive Difficulty	Within-Subject Effect	Process modeling knowledge	$22.58_{1,53}$	0.000	0.30
	Between-Subject Effect	Control-flow pattern	$1.99_{1,53}$	0.11	0.13
		Element interactivity: process structure tree distance	$17.79_{1,53}$	0.000	0.25
		Element interactivity: cut-vertex		>0.11	
		Validity of conclusion		>0.11	
	Interaction Effect	Process modeling knowledge * control-flow pattern		>0.11	
		Process modeling knowledge * process structure tree distance		>0.11	
		Process modeling knowledge * cut-vertex		>0.11	
		Process modeling knowledge * validity of conclusion		>0.11	

5.1 Process Modeling Knowledge

Concerning hypothesis 1, we observed (from Table 3) that modelers with lower modeling knowledge performed significantly worse on the deductive reasoning tasks ($F_{1,53}$=8.05, p=0.006) and rated them more difficult ($F_{1,53}$=22.58, p=0.000) than did modelers with higher process modeling knowledge. Therefore, we revealed strong evidence for hypothesis 1.

5.2 Control-Flow Patterns

Hypothesis 2 had predicted that the type of control-flow patterns that were involved in answering a question had an influence on cognitive difficulty. MANCOVA results indicated that there was, in fact, an impact of different control-flow patterns on the objective difficulty ($F_{1,53}$=3.19, p=0.02). The results further suggest that there is a trend-wise effect on subjective difficulty ($F_{1,53}$=1.99, p=0.11); albeit we note that this result is not significant at the p=0.05 level. Thus, hypothesis 1 is supported concerning objective difficulty (percentage of correct answers) but only tentatively with respect to subjective (perceived) difficulty. In addition, there is an interaction effect between experience and the difficulty of control-flow patterns. Figure 4 demonstrates that the percentage of correct answers of modelers with lower process modeling knowledge was between 6 and 11% lower than those of modelers with higher process modeling knowledge, with one exception: reasoning tasks involving *Loops*. Modelers in the “lower knowledge” group solved only 54% of those tasks correctly, whereas modelers in the “higher knowledge” group solved 78% of these tasks correctly (i.e., 24% more). For reasoning tasks that required understanding the basic *Sequence* pattern only, we obtained the smallest difference between both groups (6%).

We performed a *post-hoc* analysis (Fisher's Least Significant Difference test) to determine which types of control-flow patterns significantly differ from each other. Figure 4 and Figure 5 depict descriptive results of cognitive difficulty of control-flow patterns. First, we turn to results concerning objective difficulty. In general, tasks were most difficult if they demanded to understand *Loops,* followed by *Compound* control-flow patterns (a combination of at least two patterns other than *Sequence*), *AND* and *XOR*. Tasks for which only the control-flow pattern *Sequence* had to be understood were the easiest. *Loops* were significantly more difficult to understand than *Sequence* (Mean Diff=17.66, SD=5.21, p=0.001) and *XOR* (Mean Diff=12.33, SD=6.15, p=0.05). *AND* (Mean Diff=7.03, SD=3.16, p=0.03) and *Compound* control-flow patterns (Mean Diff=15.24, SD=4.73, p=0.002) were both more difficult to understand than *Sequence* alone.

Concerning subjective cognitive difficulty, *Compound* patterns (a combination of more than one *XOR*, *AND* and *Loop*) were most difficult; they were significantly more difficult than *Sequence* (Mean Diff=0.68, SD=0.20, p=0.001), *AND* (Mean Diff=0.52, SD=0.22, p=0.02) and *XOR* (Mean Diff=0.64, SD=0.25, p=0.01). Additionally, *Loops* were more difficult than *Sequence* (Mean Diff=0.54, SD=0.22, p=0.02).

The results also lend support to hypothesis H2a, which had predicted that the control-flow pattern *Sequence* has a lower cognitive difficulty than other control-flow patterns. H2b had predicted that reasoning tasks, for which a combination of more than one control-flow pattern (other than *Sequence*) had to be understood, were more difficult than if only a single control-flow pattern was involved. H2b was partly supported for subjective difficulty, but not supported for objective difficulty.

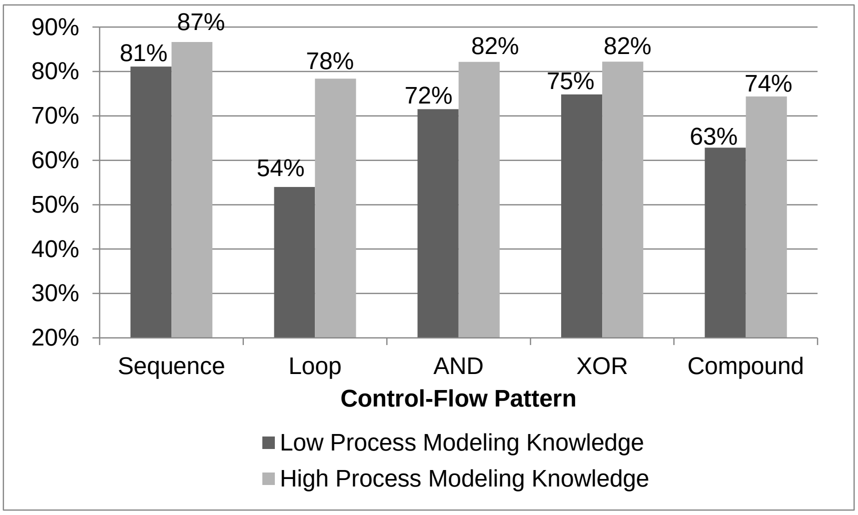


Figure 4. Different control-flow patterns and objective cognitive difficulty (percentage of correct answers).

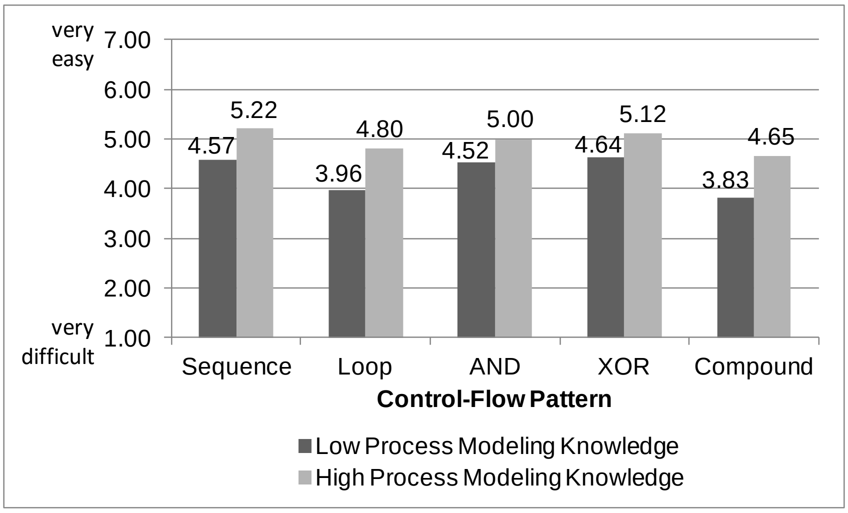


Figure 5. Different control-flow patterns and subjective cognitive difficulty (scale for perceived difficulty from 1=''very easy'' to 7=''very difficult'').

5.3 Element Interactivity

Two parameters were used to measure element interactivity: (i) the process structure tree distance and (ii) the existence of a cut-vertex. In line with our expectations, the process structure tree distance was a significant influence factor for subjective ($F_{1,53}$=17.79, p=0.000) and objective difficulty ($F_{1,53}$=22.08, p=0.000). The higher the process structure tree distance, the lower the percentage of correct answers and the higher the subjective cognitive difficulty. Figure 6 shows the percentages of correct answers, and Figure 7 provides the average results on the perceived difficulties across different process structure tree distances. However, the existence of a cut-vertex did not significantly influence cognitive difficulty. Therefore, hypothesis 3 was only partially supported.

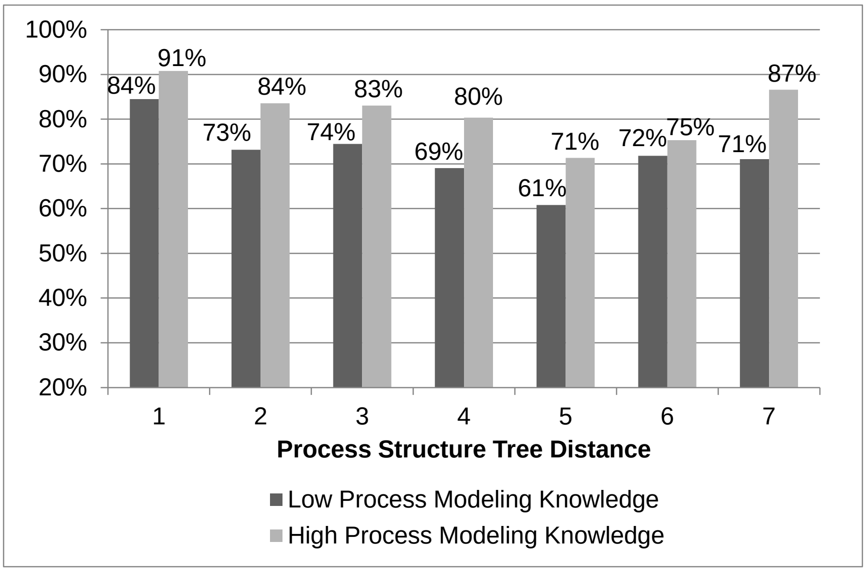


Figure 6. Process structure tree distance and objective cognitive difficulty (percentage of correct answers).

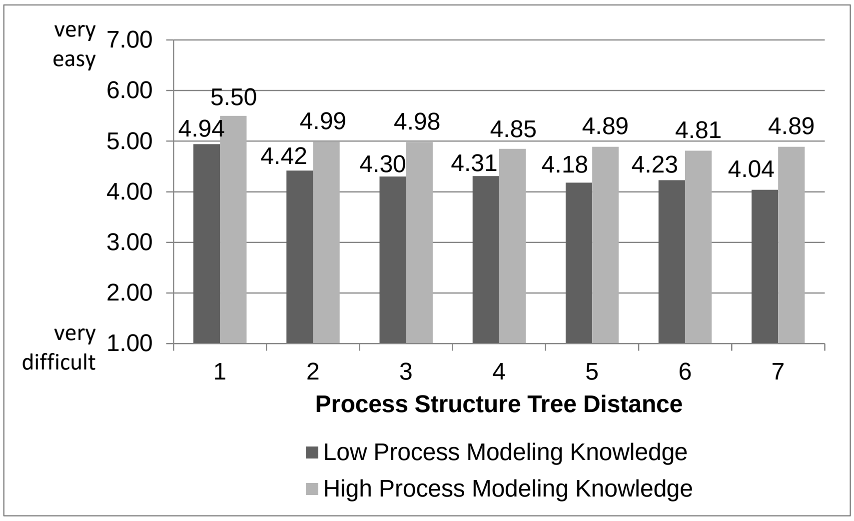


Figure 7. Process structure tree distance and subjective cognitive difficulty (scale for perceived difficulty from 1=''very easy'' to 7=''very difficult'').

5.4 Validity of Conclusion

Hypothesis 4 proposed that valid deductive reasoning tasks would be easier to answer than invalid ones. As Table 3 indicates, the validity of the conclusion did have a significant effect on objective but not on subjective difficulty. Contrary to expectations, however, we can derive from Figure 8 that valid reasoning tasks were more difficult to answer than invalid/wrong tasks. Concerning subjective difficulty, descriptive results pointed into the same unanticipated direction (see Figure 9). Thus, hypothesis 4 was not supported.

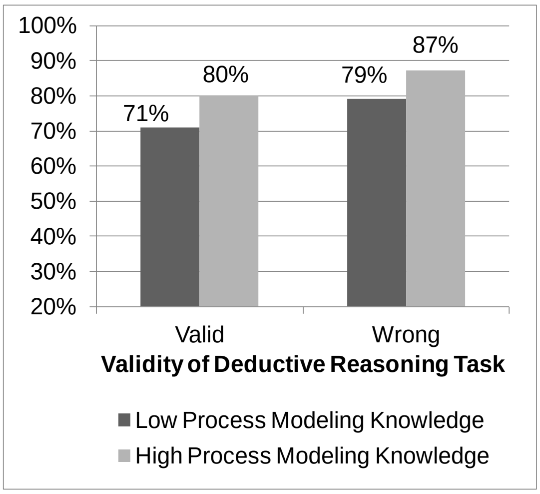


Figure 8. Validity of conclusion and objective cognitive difficulty (percentage of correct answers).

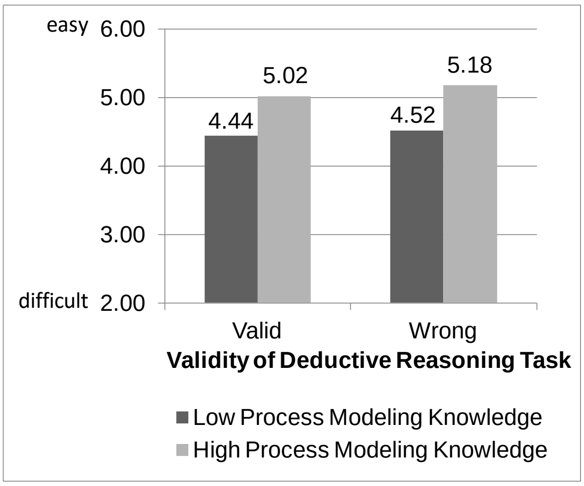


Figure 9. Validity of conclusion and subjective cognitive difficulty (scale for perceived difficulty from 1="very easy" to 7="very difficult").

Table 4 provides an overview of the results revealed regarding the hypotheses 1-4.

Table 4. Summary of hypothesis testing results.

	Independent Variable	Dependent Variable: Cognitive Difficulty	Results
H1	Process modeling knowledge	Objective	Supported
		Subjective	Supported
H2	Control-Flow Pattern	Objective	Supported
		Subjective	Supported
H2a	"*Sequence*" patterns are easier than other patterns	Objective	Supported (easier than *Loops*, *AND* and *Compound* patterns)
		Subjective	Partly Supported (easier than *Compound* patterns and *Loops*)
H2b	"*Compound*" patterns are more difficult than other patterns	Objective	Not Supported (only more difficult than *Sequence*)
		Subjective	Partly Supported (more difficult than *Sequence* and *XOR*)
H3	Element Interactivity	Objective	Supported for process structure tree distance but not for the existence of a cut-vertex
		Subjective	Supported for process structure tree distance but not for the existence of a cut-vertex
H4	Validity of conclusion	Objective	Not Supported (significant influence, but reverse effect)
		Subjective	Not Supported

6 Discussion

This study aimed at assessing the importance of influence factors for deductive reasoning on the basis of process models. We identified a number of interesting results.

First, a main finding was that deductive reasoning tasks differ in their cognitive difficulty dependent on the control-flow patterns required to answer them. In general, reasoning tasks only demanding an understanding of the control-flow pattern *Sequence* were the easiest, followed by *XOR* and *AND*, and

Compound patterns and *Loops* were the most difficult. The present finding is partly consistent with the cognitive weights proposed by Shao and Wang (2003) concerning *Sequence*, *XOR* and *Loop*. However, contrary to their proposal, we did not reveal any evidence that *AND* would be the most difficult pattern. An explanation for this discrepancy could be that parallel execution (concurrency) in programming code is harder to understand than its visualized counterpart in a process model. The findings of the current study also do not support the ideas of Sánchez-González et al. (2012), who claimed that *XORs* are more difficult to understand than *ANDs* in a process model. We could not reveal empirical evidence for a difference of cognitive difficulty between *AND* and *XOR*, which was in line with our prediction, as there do not exist strong theoretical considerations which would suggest any difference.

Second, another important finding was that element interactivity is positively related to reasoning difficulty. While we were unable to find a significant confirmation regarding the existence of a cut vertex, we obtained strong support for our hypotheses regarding a process structure tree distance. This effect has not been studied so far but is comparable to the discussion of whether the nesting level in a process model has an influence on its understandability. Mendling et al. (2007b; 2011) did not find a significant relationship between the nesting level and the understandability of a model. However, while Mendling et al. regarded the nesting level as a global attribute of a process model, we related the process structure tree distance to the model elements that we asked for. We argue that it is important to consider local parameters for assessing element interactivity. Even if a process model includes deeply nested structures, single comprehension tasks might include only neighbored elements in a specific submodel of the overall model, and thus their difficulty would not be a representative indicator for the model as a whole. As a consequence, we suggest to pay closer attention to the model parts actually relevant to answer a specific reasoning task. The rest of the complex model may have an effect on cognitive difficulty of a reasoning task. Yet, the source of cognitive load is different insofar that it mainly complicates the search and identification of relevant model structures.

While our results support the hypothesis that process structure tree distance positively influences the difficulty of a task, it remains unclear whether these results can be generalized to higher process structure tree distances. One unanticipated finding was that the approximately linear relationship between reasoning difficulty and process structure tree distance did not continue for process structure tree distances higher than five. A possible explanation for this is that the amount of reasoning tasks was lower for high process structure tree distances. This is because we had opted for process models of "average" size and complexity. We recognize that larger models would be necessary for positing reasoning tasks with higher element interactivity and for empirically determining how the relationship continues above a process structure tree distance of five. Based on theoretical considerations and empirical research on the negative effect of model size on comprehension (Mendling et al., 2007a; Mendling et al., 2010a), it is, however, likely that cognitive load increases even further for higher element interactivities.

In the following section, we want to discuss results concerning cut vertices. In contrast to our results, a similar experiment by Mendling and Strembeck (2008) provided support for the hypothesis that a process model with more cut-vertices is more easily understood, while further studies (Mendling et al., 2007b) yielded inconsistent results on this topic. In our study, 79.9% of the questions about two activities separated by a cut-vertex had been answered correctly, compared to 75.8% of the questions about two activities without a cut-vertex. While this, in fact, points to a trend into the expected direction, it was not statistically significant. However, it is still possible that, when repeating the study with a larger number of models with a cut-vertex, a significant effect could be revealed. In our study, 22 out of 61 reasoning tasks had cut vertices. Future studies regarding this question will be required.

In line with our predictions and with previous research (Mendling et al., 2012), we also found that modeling knowledge reduced the objective and subjective cognitive difficulty of reasoning on basis of a model. Modelers with higher process modeling knowledge performed better than modelers with lower process modeling knowledge in all deductive reasoning tasks. Additionally, we found that loops were especially difficult for the group with lower process modeling knowledge. A possible explanation for this result is that advanced modelers have already acquired cognitive schemas for this control structure (Détienne, 1990).

Surprisingly, the results we obtained concerning the validity of the reasoning task were contradicting our original hypotheses and indicated that wrong deductive conclusions were easier to identify than correct ones. A possible explanation for this finding might be that in some cases of wrong deductive conclusion, it might be quite obvious that it cannot hold true in the context of the model, and a falsifying argument can easily be found. In contrast, verifying a conclusion might be harder, as it demands the ruling out of all possibly falsifying arguments. This explanation is in line with research that has shown that falsification strategies are especially relevant to achieve insight into a reasoning task (Johnson-Laird and Wason, 1970). One suggestion based on this finding is that in order to improve understanding, one could provide additional information on the most important constraints and impossible process executions to a process model.

7 Limitations

In this section, we will consider the common caveats associated with laboratory experiments and specific limitations of our study, in particular possible threats to validity based on Wohlin et al. (2000).

One potential weakness of this study is the selection of subjects. We recognize that the fact that our sample was drawn from business school students might limit *external validity*. However, we believe it was more important to assure that random heterogeneity of subjects was low – which could be a potential threat to *conclusion validity* – by choosing a homogenous group of students. Moreover, the similarity of the sample to other process model comprehension studies (Figl et al., 2013a; Figl et al., 2013b; Mendling et al., 2012) eases comparison of results.

A threat regarding *external validity* related to the fact that the subjects in this study were not exposed to time constraints, which does not have to reflect the situation when process models are applied in the real world.

Concerning *internal validity*, the selection of modelers might underestimate the true relationship between process modeling knowledge and cognitive difficulty of reasoning. A selection of experts with higher practical experience might lead to even stronger performance differences between the group "less process modeling knowledge" and the group "more process modeling knowledge". However, the modelers with higher knowledge in our sample were sufficiently experienced to perform significantly better than the comparison group, and, therefore, we believe that the selection of groups was well suited to reach clear-cut results for our research questions.

Concerning *construct validity*, we like to discuss the selection of reasoning tasks. A general problem when constructing deductive reasoning tasks is that "logic allows an infinite number of different conclusions to follow validly from any premises" (Johnson-Laird, 2010). On the other hand, a specific process model with a semantic meaning sets structural constraints that limit the types of questions that can be asked. Therefore, not all combinations of reasoning tasks concerning different control flow patterns might result. To ensure that operationalizations actually measure the theoretical constructs, we used various reasoning tasks on four different models for each control-flow pattern. However, it was difficult to find reasoning tasks for which the knowledge of only one concept was not sufficient to solve (*Compound* patterns). Furthermore, in constructing the test material, we decided to use pairs of activities, which were either close (as a rough guide approximately one activity between them) or distant (>one activity between them) according to the spatio-visual distance. This variation of the location of model elements was mandatory to achieve reasoning tasks differing in "element interactivity". Still, it was difficult to find reasoning tasks with very high element interactivity, since it was also limited by the size of the models.

We opted for presenting models with textual labels in a verb-object labeling style (Leopold et al., 2013) in order to research reasoning on models which are similar to "real" business process models. However, we acknowledge that using reasoning tasks with meaningful process element labels instead of abstract labels may have lead to situations where subjects based their inference on domain knowledge instead of relying on the process model only. Also, reasoning problems might arise from problems to understand a label text. We used models from several domains (as suggested by Aranda et al. (2007)) as well as different combinations of question types and process elements with the aim to limit this effect. Replications of the study (e.g. using abstract labels or using reasoning tasks which would be answered differently based on domain knowledge than based on a given process model) can help to research the mediating context effect of textual labels on identifying deductive conclusions derived from a process model as correct or wrong.

The reader should also bear in mind that based on the possibility to guess correct answers, percentages of correctly solved reasoning tasks might be artificially increased, and therefore, should not be interpreted in an absolute sense, but only in relation to values of the same question type. While this problem cannot completely be eliminated, in order to lower guessing probability, we had also included the option “I don't know”.

8 Implications for Research

The current findings add substantially to our understanding of the cognitive difficulty of process models. In addition, our article provides a new understanding of process model comprehension questions as deductive reasoning tasks.

A lot of empirical work on the understandability of process models has been undertaken in recent years. A survey of this research field (Houy et al., 2014) shows that the most frequently used method for measuring understandability so far has been to ask questions that aim to test the comprehension of the models. Interestingly, many of the current studies have not discussed the selection of questions in detail, which seems to indicate that it is common practice to “randomly” select comprehension questions.

Melcher et al. (2010) pointed out that results obtained by such an experimental design should be validated (see also Laue and Gadatsch, 2011). They further suggested to include questions relating to different aspects of understanding the relations between activities in a model (order, concurrency, exclusiveness, and repetition). Our results strongly support this suggestion by providing empirical evidence that the selection of the model elements involved and their interrelations have an influence on the difficulty of a question. Moreover, our work adds one more perspective: As we have shown, element interactivity has an influence on understandability and interactivity should be taken into account when selecting comprehension questions. In particular, when asking questions that aim at measuring the understandability of two models, the models can be compared in a valid way only if the questions are not too divergent from one another (from an element interactivity perspective).

If only questions with a small range of element interactivity are used, the results of such an experiment cannot be generalized when reasoning about questions for which an element interactivity out of this range has to be considered. With our suggestions on the relevance of balanced selection and construction of questions, we contribute another aspect to existing guidelines for experiments on the understandability of process models (such as Patig, 2008). Another implication of our findings is that researchers need to exercise caution when interpreting existing studies in which the effect of element interactivity has not been considered when constructing comprehension questions. Consequently, replications of comprehension studies (with a more balanced set of questions) might shed light on inconsistent results and answer unresolved issues.

By introducing the concept of local complexity measures, we hope to open a new strand of research in the field of complexity metrics for business process models. While the current global complexity metrics such as those presented in Mendling et al. (2007b) can only suggest that a model might be difficult to understand, local metrics could be used to pinpoint the parts of the model that can cause understanding problems.

Furthermore, our results on differences in understanding the different control-flow patterns could also be used to adjust parameters used in global complexity metrics that build on the differences in understanding of the various building blocks of a business process model (as suggested by Lassen and van der Aalst, 2009).

Another remarkable result regarding complexity metrics is that the workflow patterns related to loops (the arbitrary cycle and structured loop pattern) seem to be more difficult to understand than the exclusive choice pattern. However, all these patterns have in common that the control flow is built from an XOR-split/XOR-join pair. We conclude that complexity metrics that just count the number of XOR gateways or the number of XOR-splits (without asking to which kind of workflow pattern the gateway belongs to, used for example in Rolon et al. (2009) and Mendling et al. (2006)), should be questioned. Our results suggest that measures such as “number of exclusive choices (which are not loop exits)” would work better than the “number of XOR-splits”.

Several research opportunities emerged from our study. Further experimental investigations could consider factors such as model size variations, abstract versus concrete labels and high practical modeling experience in order to establish an integrative understanding of determinants of deductive reasoning performance. It would be interesting to include comprehension parameters as time needed to answer each reasoning task (Houy et al., 2012) in a computer-based assessment. Additional studies could validate our results based on a larger number and variety of deductive reasoning tasks. As fellow scholars have already collected a variety of data sets with answers on process-model comprehension questions, we suggest re-analyzing these data sets on item level with the research questions presented in this paper.

9 Practical Implications

The presented findings are of direct practical relevance. When we better understand the influence of cognitive load on the comprehensibility of a model, it will be possible to manage cognitive load and – in the end – to obtain models that are more helpful for communication. This has implications for business process modeling practice and education and for the potential to promote acceptance and use of process models in organizations.

Reducing unnecessary cognitive load can help to make the process model more understandable. Cognitive Load Theory suggests that the information presented in a model should be structured in a way that the reader can reduce the cognitive load by assimilating groups of model elements. Our work provides empirical evidence that high interactivity of elements may improve cognitive load and lower comprehensibility of process models. Thus, we further encourage model tool designers to provide options for syntax highlighting. (Reijers et al., 2011) describe a method to use different colors for identifying split-join pairs belonging together. This provides users with a visual cue to improve comprehensibility of deeply nested blocks.

To support the modeler, it has been suggested that model editors calculate (global) complexity metrics and warn the modeler when they exceed a certain threshold (Sánchez-González et al., 2012). However, even more useful than reporting the overall complexity is to inform the modeler *which* parts of the model are likely to raise comprehension problems. By introducing the concept of local complexity measures, we step further into this direction. A model editor could then highlight the parts of a model that might impose difficulties.

Visualizing the nesting level of software has been found to be a useful support for computer programmers (Ball and Eick, 1996). It can serve as a comprehension aid and as an indicator for code elements that are difficult to understand. For textual documents, there are tools that measure and visualize complexity and suggest how the readability can be improved (Newbold and Gillam, 2008). In a similar way, we think that it could be helpful to highlight the parts of a process model that increase the interactivity between the elements in the model.

However, tool support does not have to be restricted to locating parts of the model that can be difficult to understand. It is also feasible that the modeling tool suggests replacing this part by a more comprehensible behavior-equivalent model variant. Patterns for model modifications that aim to improve the understandability while preserving the behavior of a model have been described by La Rosa et al. (2011). The authors of this work discuss that "automated support to suggest [a complexity reducing pattern] for increasing the understandability is missing in the current generation of process model editors." The concept of local model complexity can spur new generations of such editors which detect the parts of a model that can impose understandability problems and suggest improvements.

Future research is needed to determine valid and reliable values for the cognitive difficulty of understanding specific relations model fragments (control-flow patterns as for instance sequence or loop). These values could finally rate the understandability of models without the need for user evaluation. Looking ahead, exact comprehension values could then be used to guide modeling tool developers to provide feedback on the cognitive difficulty of models to users.

Last but not least, the percentage of wrong answers given by modelers with lower process modeling knowledge to questions related to the different control-flow patterns allows for conclusions on how to teach business process modeling. In particular, it seems to be important to discuss non-trivial models with loops. Also, we recommend providers of modeling trainings and lecturers to draw their attention to teach students of the constraints of information processing in human memory and their implications for understanding models.

10 Conclusion

This study is the first experimental analysis of influence factors on the cognitive difficulty of deductive reasoning tasks on the basis of a process model. Our work is an extension of existing literature, which has predominantly looked at global understandability of process models. With reference to the hypotheses posed at the beginning of this study, we can now state that interactivity of elements involved in a reasoning task is related to cognitive difficulty. Another major finding was that the difficulty of control-flow patterns varies with *Loops* as well as combinations of at least two patterns other than *Sequence* that are more difficult than *XOR* and *AND*. Modelers with lower process modeling knowledge perform worse on deductive reasoning tasks, and results suggest that they have specific difficulties with tasks including loops. It was also shown that it is easier to correctly identify a wrong conclusion than to verify a correct conclusion drawn on the basis of a model. Our work, moreover, assists in the understanding of possible comprehension problems in process models and can guide modeling tool developers to provide adequate feedback on the cognitive difficulty of model parts.

From a more general perspective, our research serves as an initial contribution on human understanding of process logic concepts (notably parallel execution, loops and decisions), which can also be classified as fundamental ideas of the computer science discipline (Zendler and Spannagel, 2008) and represent a subset of humans' "computational thinking" (Wing, 2008). Identifying cognitive difficulties in comprehending processes may ultimately lead to a better understanding of differences in how humans and computers "think".

11 References

Aranda, J., Ernst, N., Horkoff, J., Easterbrook, S.M., 2007. A Framework for Empirical Evaluation of Model Comprehensibility, 29th International Conference on Software Engineering (ICSE'07). International Workshop on Modeling in Software Engineering (MiSE-07).

Baddeley, A.D., 1992. Working memory. Science 255, 556-559.

Ball, T., Eick, S.G., 1996. Software visualization in the large. Computer 29, 33-43.

Bateson, A.G., Alexander, R.A., Murphy, M.D., 1987. Cognitive processing differences between novice and expert computer programmers. International Journal of Man-Machine Studies 26, 649-660.

Batra, D., Davis, J.G., 1992. Conceptual Data Modelling in Database Design: Similarities and Differences between Expert and Novice Designers. International Journal of Man-Machine Studies 37, 83-101.

Beller, S., Spada, H., 2003. The logic of content effects in propositional reasoning: The case of conditional reasoning with a point of view. Thinking & Reasoning 9, 335-378.

Braine, M., O'Brien, D., Noveck, I., Samuels, M., Lea, B., Fisch, S., Yang, Y., 1995. Predicting Intermediate and Multiple Conclusions in Propositional Logic Inference Problems: Further Evidence for a Mental Logic. Journal of Experimental Psychology: General 124, 263-292.

Burton-Jones, A., Wand, Y., Weber, R., 2009. Guidelines for Empirical Evaluations of Conceptual Modeling Grammars. Journal of the Association for Information Systems 10, 495-532.

Cant, S.N., Jeffery, D.R., Henderson-Sellers, B., 1995. A conceptual model of cognitive complexity of elements of the programming process. Information and Software Technology 37, 351-362.

Cater, S.C., Iyengar, S.S., Fuller, J., 1984. Computation of logical effort in high level languages. Computer Languages 9, 133-148.

Curtis, B., Kellner, M.I., Over, J., 1992. Process Modeling. Communications of the ACM 35, 75-90.

Damer, T.E., 2013. Attacking Faulty Reasoning: A Practical Guide to Fallacy-Free Arguments, 7 ed. Wadsworth Publishing, Boston, USA.

Davies, S.P., 1994. Knowledge restructuring and the acquisition of programming expertise. International Journal of Human-Computer Studies 40, 703-726.

De Neys, W., Schaeken, W., d'Ydewalle, G., 2005. Working memory and everyday conditional reasoning: Retrieval and inhibition of stored counterexamples. Thinking and Reasoning 11, 349-381.

Détienne, F., 1990. Expert Programming Knowledge: a Schema-Based Approach, in: Hoc, J.-M., Green, T.R.G., samurcay, R., Gilmore, D. (Eds.), Psychology of Programming. Academic Press, People and Computer Series, pp. 205-222.

Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A., 2013. Fundamentals of Business Process Management. Springer, Berlin, Germany.

Evans, J.S.B.T., 1972. On the problems of interpreting reasoning data: Logical and psychological approaches. Cognition 1, 373-384.

Figl, K., Mendling, J., Strembeck, M., 2013a. The Influence of Notational Deficiencies on Process Model Comprehension. Journal of the Association for Information Systems 14, 312-338.

Figl, K., Recker, J., Mendling, J., 2013b. A study on the effects of routing symbol design on process model comprehension. Decision Support Systems 54, 1104-1118.

Gemino, A., Wand, Y., 2004. A Framework for Empirical Evaluation of Conceptual Modeling Techniques. Requirements Engineering 9, 248-260.

Gilmore, D.J., 1990. Expert Programming Knowledge: A Strategic Approach, in: Hoc, J.-M., Green, T.R.G., Samurcay, R., Gilmore, D. (Eds.), Psychology of Programming, People and Computer Series, pp. 223-234.

Green, T.M., Ribarsky, W., Fisher, B., 2009. Building and applying a human cognition model for visual analytics. Information Visualization 8, 1-13.

Guceglioglu, A.S., Demirors, O., 2005. Using Software Quality Characteristics to Measure Business Process Quality, in: van der Aalst, W.M.P., Benatallah, B., Casati, F., Curbera, F. (Eds.), Business Process Management. Springer Berlin / Heidelberg, pp. 374-379.

Hoc, J.-M., Green, T.M., Samurçay, R., Gilmore, D.J., 1990. Psychology of Programming. European Association of Cognitive Ergonomics and Academic Press.

Houy, C., Fettke, P., Loos, P., 2012. Understanding Understandability of Conceptual Models – What Are We Actually Talking about?, in: Atzeni, P., Cheung, D., Ram, S. (Eds.), Conceptual Modeling. Springer Berlin Heidelberg, pp. 64-77.

Houy, C., Fettke, P., Loos, P., 2014. On the theoretical foundations of research into the understandability of business process models, Twenty Second European Conference on Information Systems, Tel Aviv.

Johnson-Laird, P.N., 2010. Deductive reasoning. Wiley Interdisciplinary Reviews: Cognitive Science 1, 8-17.

Johnson-Laird, P.N., Wason, P.C., 1970. A theoretical analysis of insight into a reasoning task. Cognitive Psychology 1, 134-148.

La Rosa, M., Wohed, P., Mendling, J., ter Hofstede, A.H.M., Reijers, H.A., van der Aalst, W.M.P., 2011. Managing Process Model Complexity via Abstract Syntax Modifications IEEE Transactions on Industrial Informatics 7, 614-629.

Lassen, K.B., van der Aalst, W.M.P., 2009. Complexity metrics for Workflow nets. Information and Software Technology 51, 610-626.

Laue, R., Gadatsch, A., 2011. Measuring the Understandability of Business Process Models - Are We Asking the Right Questions?, Business Process Management Workshops. Lecture Notes in Business Information Processing. Springer, Berlin Heidelberg, pp. 37-48.

Leopold, H., Eid-Sabbagh, R.-H., Mendling, J., Azevedo, L.G., Baião, F.A., 2013. Detection of naming convention violations in process models for different languages. Decision Support Systems 56, 310-325.

Marcus, N., Cooper, M., Sweller, J., 1996. Understanding Instructions. Journal of Educational Psychology 88, 49-63.

Markovits, H., Doyon, C., Simoneau, M., 2002. Individual differences in working memory and conditional reasoning with concrete and abstract content. Thinking & Reasoning 8, 97-107.

Matlock-Hetzel, S., 1997. Basic Concepts in Item and Test Analysis, Annual Meeting of the Southwest Educational Research Association, Austin.

Melcher, J., Mendling, J., Reijers, H.A., Seese, D., 2010. On Measuring the Understandability of Process Models, Business Process Management Workshops. Lecture Notes in Business Information Processing. Springer, Berlin Heidelberg, pp. 465-476.

Mendling, J., Moser, M., Neumann, G., Verbeek, H.M.W., van Dongen, B.F., van der Aalst, W.M.P., 2006. A Quantitative Analysis of Faulty EPCs in the SAP Reference Model.

Mendling, J., Neumann, G., van der Aalst, W.M.P., 2007a. Understanding the occurrence of errors in process models based on metrics, OTM Conferences. Lecture Notes in Computer Science, pp. 113-130.

Mendling, J., Reijers, H., van der Aalst, W.M.P., 2010a. Seven process modeling guidelines (7PMG). Information and Software Technology 52.

Mendling, J., Reijers, H.A., Cardoso, J., 2007b. What makes process models understandable. In Gustavo Alonso, Peter Dadam, and Michael Rosemann, editors, Business Process Management: Proceedings of the 5th International Conference BPM pages, 48-63.

Mendling, J., Reijers, H.A., Recker, J., 2010b. Activity labeling in process modeling : Empirical insights and recommendations. Information Systems 35, 467-482.

Mendling, J., Strembeck, M., 2008. Influence Factors of Understanding Business Process Models, in: Abramowicz, W., Fensel, D. (Eds.), Business Information Systems - BIS 2008. Springer, pp. 142-153.

Mendling, J., Strembeck, M., Recker, J., 2012. Factors of Process Model Comprehension — Findings from a Series of Experiments. Decision Support Systems 53, 195-206.

Naess, A., 1961. A Study of 'Or'. Synthese 13, 49--60.

Newbold, N., Gillam, L., 2008. Towards Automatic Document Quality Assurance, Workshop on Natural Language Processing resources, algorithms and tools for authoring aids. Language Resources and Evaluation Conference, Marrakech, Morocco, p. 27.

Object Management Group, 2013. Business Process Model and Notation (BPMN) Version 2.0.2.

Patig, S., 2008. A practical guide to testing the understandability of notations, Proceedings of the fifth Asia-Pacific conference on Conceptual Modelling-Volume 79. Australian Computer Society, Inc., pp. 49-58.

Petre, M., 1995. Why looking isn't always seeing: readership skills and graphical programming. Commun. ACM 38, 33-44.

Pichler, P., Weber, B., Zugal, S., Pinggera, J., Mendling, J., Reijers, H.A., 2012. Imperative versus Declarative Process Modeling Languages: An Empirical Investigation, Business Process Management Workshops. Lecture Notes in Business Information Processing., pp. 383-394.

Preacher, K., Rucker, D., MacCallum, R., Nicewander, W., 2005. Use of the Extreme Groups Approach: A Critical Reexamination and New Recommendations. Psychol Methods 10, 178-192.

Recker, J., Safrudin, N., Rosemann, M., 2012. How Novices Design Business Processes. Information Systems 37, 557-573.

Reijers, H.A., Freytag, T., Mendling, J., Eckleder, A., 2011. Syntax Highlighting in Business Process Models. Decision Support Systems 51, 339-349.

Reijers, H.A., Mendling, J., 2011. A Study into the Factors that Influence the Understandability of Business Process Models. IEEE Transactions on Systems, Man, and Cybernetics - Part A 41, 449-462.

Rinderle, S., Reichert, M., Dadam, P., 2004. On Dealing with Structural Conflicts between Process Type and Instance Changes, in: Desel, J., Pernici, B., Weske, M. (Eds.), Business Process Management. Springer Berlin Heidelberg, pp. 274-289.

Rist, R.S., 1989. Schema creation in programming. Cognitive Science 13, 389-414.

Rolón, E., Sanchez, L., Garcia, F., Ruiz, F., Piattini, M., Caivano, D., Visaggio, G., 2009. Prediction Models for BPMN Usability and Maintainability, Conference on Commerce and Enterprise Computing. IEEE Computer Society, pp. 383-390.

Rosemann, M., 2006. Potential pitfalls of process modeling: part A. Business Process Management Journal 12, 249-254.

Sánchez-González, L., García, F., Ruiz, F., Mendling, J., 2012. Quality indicators for business process models from a gateway complexity perspective. Information and Software Technology 54, 1159-1174.

Shanks, G., 1997. Conceptual Data Modelling: An Empirical Study of Expert and Novice Data Modellers. Australasian Journal of Information Systems 4, 63-73.

Shao, J., Wang, Y., 2003. A new measure of software complexity based on cognitive weights. Canadian Journal of Electrical and Computer Engineering 28, 69-74.

Siegmund, J., Schumann, J., 2014. Confounding parameters on program comprehension: a literature survey. Empirical Software Engineering, 1-34.

Süß, H.-M., Oberauer, K., Wittmann, W.W., Wilhelm, O., Schulze, R., 2002. Working-memory capacity explains reasoning ability—and a little bit more. Intelligence 30, 261-288.

Sweller, J., 1994. Cognitive load theory, learning difficulty, and instructional design. Learning and Instruction 4, 295-312.

Tegarden, D.P., Sheetz, S.D., Monarchi, D.E., 1995. A software complexity model of object-oriented systems. Decis. Support Syst. 13, 241-262.

Umphress, D.A., Hendrix, T.D., Cross Ii, J.H., Maghsoodloo, S., 2006. Software visualizations for improving and measuring the comprehensibility of source code. Science of Computer Programming 60, 121-133.

van der Aalst, W.M.P., ter Hofstede, A.H.M., Kiepuszewski, B., Barros, A.P., 2003. Workflow Patterns. Distrib. Parallel Databases 14, 5-51.

Vanderfeesten, I., Reijers, H.A., Aalst, W.M.P.v.d., 2008. Evaluating workflow process designs using cohesion and coupling metrics. Computers in Industry 59, 420-437.

Vanhatalo, J., Völzer, H., Koehler, J., 2009. The refined process structure tree. Data & Knowledge Engineering 68, 793 - 818.

Weitlaner, D., Guettinger, A., Kohlbacher, M., 2013. Intuitive Comprehensibility of Process Models, in: Fischer, H., Schneeberger, J. (Eds.), S-BPM ONE - Running Processes. Springer Berlin Heidelberg, pp. 52-71.

Wetherick, N.E., Gilhooly, K.J., 1995. 'Atmosphere', matching, and logic in syllogistic reasoning. Current Psychology 14, 169-178.

Wing, J.M., 2008. Computational thinking and thinking about computing. Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences 366, 3717-3725.

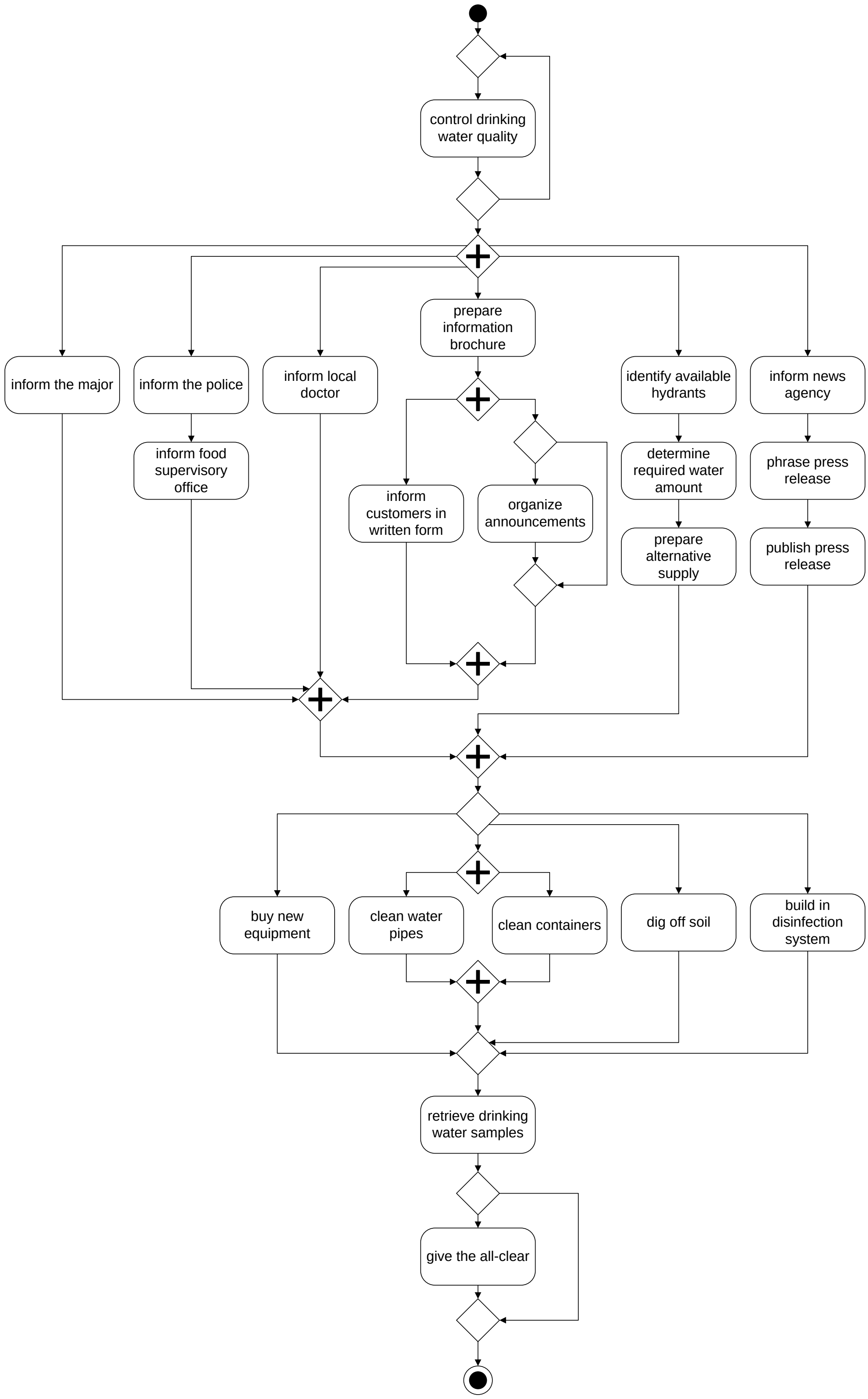
Wohlin, C., Runeson, P., Höst, M., Ohlsson, M., Regnell, B., Wesslén, A., 2000. Experimentation in Software Engineering – An Introduction. Kluwer Academic Publishers, Boston MA.

Woodworth, R.S., Sells, S.B., 1935. An atmosphere effect in formal syllogistic reasoning. Journal of Experimental Psychology 18, 451-460.

Yang, J., Hendrix, T.D., Chang, K.H., Umphress, D., 2005. An empirical validation of complexity profile graph, Proceedings of the 43rd annual Southeast regional conference - Volume 1. ACM, Kennesaw, Georgia, pp. 143-149.

Zendler, A., Spannagel, C., 2008. Empirical Foundation of Central Concepts for Computer Science Education. J. Educ. Resour. Comput. 8, 1-15.

1. Appendix A: Example of Experimental Material: Model and Characteristics of Reasoning Task



			Control Flow Pattern	Element Interactivity		Validity of Conclusion
Item Wording	Version	Reasoning Task		Process Structure-Tree Distance	Cut Vertex	
“at the same point of time”	A	"Determine required water amount" and "inform the major" can be executed at the same point of time.	AND	2	No	Correct
“at the same point of time”	B	"Inform news agency" and "phrase press release" can be executed at the same point of time.	Sequence	1	No	Wrong
“in parallel”	A	"Inform local doctor" and "publish press release" can be executed in parallel.	AND	2	No	Correct
“in parallel”	B	"Prepare alternative supply" and "publish press release" can be executed in parallel.	AND	3	No	Correct
“as well as”	A	In one process instance "buy new equipment" as well as "dig off soil" can be executed.	XOR	1	No	Wrong
“as well as”	B	In one process instance "clean water pipes" as well as "build in disinfection system" can be executed.	XOR	2	No	Wrong
“mutually exclusive”	A	The process steps "build in disinfection system" and "clean water pipes" are mutually exclusive.	XOR	2	No	Correct
“mutually exclusive”	B	The process steps "dig off soil" and "buy new equipment" are mutually exclusive.	XOR	1	No	Correct
“more often than”	A	"Control drinking water quality" can be executed more often than "identify available hydrants".	Compound	3	Yes	Correct
“more often than”	B	"Identify available hydrants" can be executed more often than "clean containers".	Compound	5	Yes	Correct
“exactly as often as”	A	In each process instance "clean containers" is executed exactly as often as "retrieve drinking water samples".	XOR	3	No	Wrong
“exactly as often as”	B	In each process instance "organize announcements" is executed exactly as often as "control drinking water quality".	Compound	6	Yes	Wrong
“is executed before”	A	If "inform the police" as well as "inform food supervisory office" are executed in a process instance, then "inform the police" is executed before "inform food supervisory office".	Sequence	1	No	Correct
“is executed before”	B	If "inform the police" as well as "give the all-clear" are executed in a process instance, then "give the all-clear" is executed before "inform the police".	Sequence	4	Yes	Wrong
“has to be finalized before”	A	If "prepare information brochure" as well as "organize announcements" are executed in a process instance, then "organize announcements" has to be finalized before "prepare information brochure" can start.	Sequence	3	No	Wrong
“has to be finalized before”	B	If "prepare information brochure" as well as "retrieve drinking water samples" are executed in a process instance, then "prepare information brochure" has to be finalized before "retrieve drinking water samples" can start.	Sequence	3	Yes	Correct