Information and Software Technology 87 (2017) 81–102

Contents lists available at ScienceDirect

Information and Software Technology

journal homepage: www.elsevier.com/locate/infsof

Investigating styles in variability modeling: Hierarchical vs. constrained styles

Iris Reinhartz-Berger[a,*], Kathrin Figl[b], Øystein Haugen[c]

[a] Department of Information Systems, University of Haifa, Israel
[b] Institute for Information Systems and New Media, Vienna University of Economics and Business, Vienna, Austria
[c] Østfold University College, Halden, Norway

ARTICLE INFO

Article history:
Received 28 April 2016
Revised 26 January 2017
Accepted 27 January 2017
Available online 2 February 2017

Keywords:
Variability modeling
Feature modeling
Comprehensibility
Hierarchical modeling
Textual constraints
Cognitive aspects
Empirical research
Product line engineering

ABSTRACT

Context: A common way to represent product lines is with variability modeling. Yet, there are different ways to extract and organize relevant characteristics of variability. Comprehensibility of these models and the ease of creating models are important for the efficiency of any variability management approach.
Objective: The goal of this paper is to investigate the comprehensibility of two common styles to organize variability into models – *hierarchical* and *constrained* – where the dependencies between choices are specified either through the hierarchy of the model or as cross-cutting constraints, respectively.
Method: We conducted a controlled experiment with a sample of 90 participants who were students with prior training in modeling. Each participant was provided with two variability models specified in Common Variability Language (CVL) and was asked to answer questions requiring interpretation of provided models. The models included 9–20 nodes and 8–19 edges and used the main variability elements. After answering the questions, the participants were asked to create a model based on a textual description.
Results: The results indicate that the hierarchical modeling style was easier to comprehend from a subjective point of view, but there was also a significant interaction effect with the degree of dependency in the models, that influenced objective comprehension. With respect to model creation, we found that the use of a constrained modeling style resulted in higher correctness of variability models.
Conclusions: Prior exposure to modeling style and the degree of dependency among elements in the model determine what modeling style a participant chose when creating the model from natural language descriptions. Participants tended to choose a hierarchical style for modeling situations with high dependency and a constrained style for situations with low dependency. Furthermore, the degree of dependency also influences the comprehension of the variability model.

1. Introduction

Variability management is essential when dealing with similar complex systems. We need to manage the variability for several different reasons, such as proper test coverage, flexible product portfolio, high degree of reuse, and necessary adaptation to a changing environment. In order to make the development process effective and efficient in these cases, reuse needs to be done systematically and not ad-hoc. To this end, the similarities as well as the differences among the systems have to be analyzed and represented in some comprehensible way for the various stakeholders involved in the development process.

Various approaches to variability modeling have been suggested over the years. Among those we can mention feature modeling, orthogonal variability modeling, UML-based variability modeling, and decision modeling. *Feature modeling* in general [12] and Feature-Oriented Domain Analysis (FODA) [38] in particular promotes representing variability in feature models, which are graphs or trees that describe end-user visible characteristics (features) of systems in a product line, illustrating the relationships and constraints (dependencies) between them. *Orthogonal variability modeling* suggests specifying variability in separate models, which are linked to the development artifacts, termed base models. Examples of languages in this category are Orthogonal Variability Models (OVM) [53] and Common Variability Language (CVL) [31]. CVL with its aspiration to become a standard for variability modeling could simulate feature models and OVM models, but used different terms. Features of a feature model would correspond to choices in CVL. The third category of *UML-based variability*

* Corresponding author.
E-mail addresses: iris@is.haifa.ac.il (I. Reinhartz-Berger), kathrin.figl@wu.ac.at (K. Figl), oystein.haugen@hiof.no (Ø. Haugen).

http://dx.doi.org/10.1016/j.infsof.2017.01.012

modeling, which includes, for example, PLUS [29] and ADOM [58], extends UML metamodel or introduces profiles with stereotypes to describe variability-related terms, such as mandatory (kernel), optional, variation point, and variant. Finally, *decision modeling* is based on representation of decisions that "are adequate to distinguish among the members of an application engineering product family and to guide adaptation of application engineering work products" [16, p. 174]. As opposed to feature modeling which focuses on domain representation, decision modeling emphasizes product derivation.

Even after choosing a specific variability modeling language, different models can be created to represent the same variability (i.e., set of differences). These models may differ in the characteristics (choices) they contain or the ways in which these choices are organized. We examine two common ways to represent variability: *hierarchical*, where the dependencies or constraints between choices are implicitly specified through the hierarchy of the model, and *constrained*, where the dependencies are explicitly specified as constraints (expressed textually or via visual edges).[1] We use the term "modeling style" to refer to these two types of variability representation. This is in line with the way the term modeling style is defined and used in other contexts, for instance in a style book on UML: "a standard would involve using a squared rectangle to model a class on a class diagram, whereas a style would involve placing subclasses on diagrams below their superclass(es)" [2, p. 2]. Note that we are not comparing notations but concentrate on the modeling style. We apply CVL [31], but we could have used another notation for variability modeling to fulfill the same objective. Galster et al. [26] refer to many studies where variability descriptions are applied, but they do not mention any studies where different styles of representation have been empirically compared for comprehension.

To demonstrate differences in style, consider the two models in Fig. 1, which specify the variability within basic choices of Skoda Yeti cars. Both models use CVL notation. The figure labeled (a) follows a hierarchical modeling style, constraining, for example, active diesel cars to be manual. Note that this modeling style results in repetition of choices, but repetition of choices in variability models is already acknowledged by concepts such as "feature reference" [18]. The figure labeled (b), on the other hand, specifies the fuel-, gear-, drive-, and gadget level-related characteristics in separate branches (although hierarchically in the form of a tree) and the dependencies between these characteristics are specified as textual constraints. Thus, we consider it as following the constrained modeling style.

The selection of the modeling style may influence the comprehension of variability and consequently the effectiveness and efficiency of variability management. These aspects are relevant, regardless of whether the variability models are created manually (by humans) or automatically (by generators or reverse engineering tools).

Prior research has investigated how different variability modeling notations may affect comprehension. In [56], the comprehensibility of two orthogonal variability modeling methods – CVL and OVM – has been evaluated in terms of understanding variability models and their relations to the development artifacts. In [59], the comprehensibility of a feature-oriented notation (CBFM) and a UML-based variability modeling method (ADOM) has been compared for different stakeholders (developers and customers/end users). In [57] the comprehensibility of CVL models to participants familiar and unfamiliar with feature modeling has been examined. These studies focus on different modeling notations or relevant stakeholders. They have not addressed the effect of alternative ways (modeling styles) to represent variability models after a specific notation and type of stakeholders have been selected. To fill this gap, our aim is to investigate what the benefits and limitations of each modeling style are on the comprehension of variability. We particularly refer to comprehension in both interpreting (reading) and creating (writing) models. Czarnecki and Wasowski [19] have already mentioned the importance of considering human cognitive limits for choosing a representation. However, to the best of our knowledge, no empirical study has been conducted on the aforementioned modeling styles in related areas, including software engineering and conceptual modeling.

Our research can be characterized as intragrammar evaluation [27], as it compares different ways to apply the grammar (CVL) and, in doing so, investigates "principles for improving the use of one grammar when used on its own" [10, p. 39]. The main contributions of this paper are to pinpoint when the different styles are best applied and what the consequences of the different styles are on comprehension.

The rest of the paper is organized as follows. Section 2 provides an overview of the theoretical and technical background relevant to the research. We outline the research framework and hypotheses in Section 3 and then describe in Section 4 the design of the experiment we used to test our propositions. Section 5 presents our data analysis and the findings of the research. In Section 6, we report and discuss the results, and in Section 7 the implications for research and practice, as well as the threats to validity, are presented. Finally, Section 8 summarizes the findings and outlines future research directions.

2. Theoretical and technical background

In this section we provide the theoretical background, elaborating on representation of things and properties in conceptual modeling (Section 2.1), variability modeling styles and their properties (Section 2.2), and cognitive effectiveness of variability modeling styles (Section 2.3). We further provide the necessary technical background on variability spaces and CVL in Section 2.4.

2.1. Representation of things and properties

There is a long tradition of research on how to model things and properties. Features, attributes, and properties are central to most theories that deal with how humans build classification categories of concepts. For instance, *defining features* can uniquely identify a category as a necessary attribute; *characteristic features* may describe prototypes; or humans may be aware of *essential, incidental,* and *accidental features* to build a complex mental theory of concepts [68].

In the context of domain modeling, researchers have predominantly investigated the effect of alternative representations of properties in Entity Relationship (ER) diagrams [8,9,28,49] and UML diagrams [10,64] on users' domain understanding. Most of these works theoretically build on the Bunge–Wand–Weber (BWW) framework [75,76,77] and good decomposition models that adapt ontological theory to conceptual modeling.

In contrast to domain modeling, variability modeling has a stronger focus on "identifying commonality and variability in a domain" rather than "differentiating concepts from features" or "describing all details of products" [44, p. 65]. Czarnecki et al. [17] categorize feature modeling as a "notational subset of ontologies" or as a specific view on ontologies. Asadi et al. [3] suggest a mapping of variability concepts to the BWW framework. Specifically, they claim that features refer to natural kind, which is "a kind of things adhering to the same laws." Based on this mapping, they

[1] Note, some variability models, such as feature diagrams and CVL models, are always structured hierarchically. Hence, by constrained modeling style we refer to situations in which dependencies or restrictions are expressed through constraints and not through the diagram hierarchy.

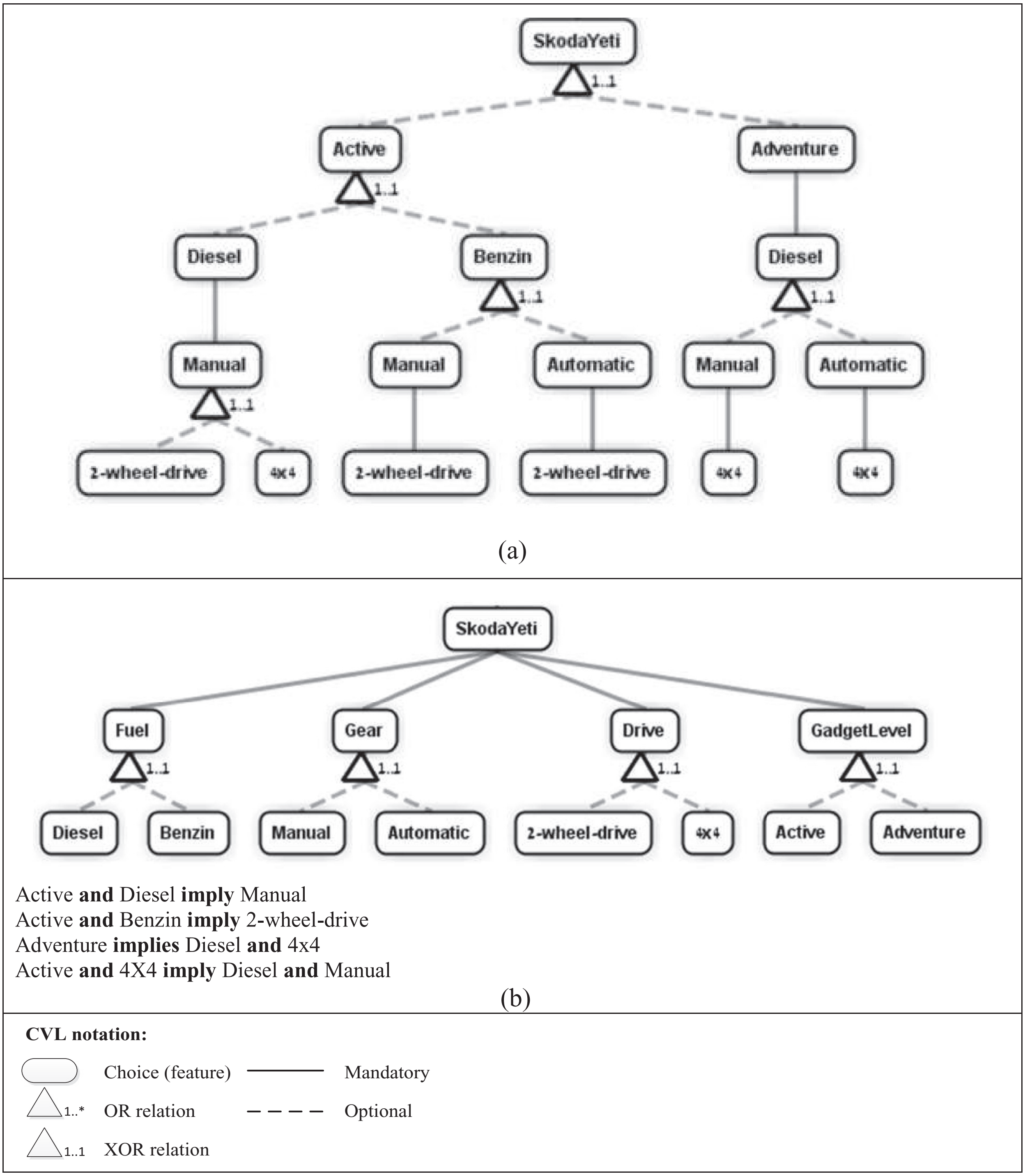

Fig. 1. CVL models specifying the variability within basic choices of Skoda Yeti cars: (a) hierarchical style and (b) constrained style.

further derive variability patterns and analyze how existing variability modeling languages support these types of variability. Their analysis is intergrammatical, as it mainly focuses on two variability modeling languages – feature models and OVM [53]. We, on the other hand, concentrate in this work on *intra*grammatical aspects in the form of modeling styles.

2.2. Variability modeling styles and their properties

The extraction and representation of variability models are the focus of many studies dealing with reverse engineering from source code, configurations, or requirements, e.g., [1,65]. Given the same input, these studies usually generate a single variability model [34], although different models may exist for the same case [19]. As noted in the introduction, these models may differ in the ways choices are structured.

Moody [48, p. 766] claims that “to effectively represent complex situations, visual notations must provide mechanisms for modularization and hierarchically structuring.” *Modularization* supports dividing large systems into smaller parts or subsystems in order to reduce complexity. Supported in cognitive load theory, this mechanism may “improve speed and accuracy of understanding” and "facilitate deep understanding of information content." *Hierarchy*, on the other hand, supports top down understanding and enables controlling complexity by organizing elements at different levels of detail, "with complexity manageable at each level." From a cognitive point of view, the 'framework for assessing hierarchy' by Zugal et al. [78] gives a clear account of possible effects of hierarchy in visual models on the mental effort: while 'abstraction' decreases mental effort due to information hiding and pattern recognition, 'fragmentation' increases mental effort, because users have to switch between fragments and integrate information.

Many variability models, such as feature diagrams and CVL models, represent a hierarchical structure to a certain degree. We are less concerned on *whether* to use hierarchical structuring or not, but *how* to represent variability. Since there are different ways to model variability, effort has to be put into understanding the strengths and the weaknesses of different modeling styles. In the context of this paper, we are not interested in automatically inferring possible configurations, but on inferring configurations in the model reader's mind. As noted, this paper explores two common modeling styles in the context of variability representation: hierarchical and constrained.

Both modeling styles require some kind of classification to organize the choices in a tree hierarchy. In general, classification serves two purposes: cognitive economy and support of inferences [51]. Applied to the variability modeling domain, this means that the resource reduction effect compared to a list of all possible configurations as well as the easiness with which correct configurations can be inferred from the model determines the cognitive effectiveness of a variability model. The selection of elements (choices) for a variability model should therefore balance these two goals [52]. Parsons and Wand [52, p. 253] refer to two main principles to reach that goal in class models: *completeness* ("All relevant information about each phenomenon (instance) in a domain should be included") and *efficiency* ("Minimize resources used in maintaining and processing information"). The efficiency principle includes "non-redundancy" because redundancy "might require additional resources in maintenance and retrieval and hence will violate the principle of cognitive economy" [51, p. 6]. In variability modeling, redundancy can occur due to repetition of choices to constrain possible configurations (see, for example, "$4 \times 4''$ in the model depicted in Fig. 1(a) and the same element in the graphical model as well as the three last constraints specified in Fig. 1(b)).

Generally, although the repetition of choices is intuitive, it is not obvious how redundancy should be formally treated. Batory [5] has explicitly excluded repetitions, but Czarnecki and Kim [18] enable some kind of repetition by introducing the concept of "feature reference" to increase reuse and support scalability. Repetition of concepts (nodes) in tree structures has been proven to be efficient in terms of human comprehension for other purposes including, e.g., decision trees [61] and logic trees. It has recently been described that repeating choices represent a language challenge since the repeated choices obviously represent something common, while the repetition shows that there are structural differences related to the choices [32]. In particular, the challenge becomes evident when repeated choices also appear in explicit constraints. Since the intuition is quite clear in these cases, our study does not have to deal with the formal interpretation of repeated choices.

In the same vein, Czarnecki and Wasowski [19] refer to two properties to create models in the area of automatic feature extraction: (1) *maximality*: "the resulting feature model graphically exposes maximum logical structure" and (2) *minimality*: "the resulting feature model avoids redundancy in the representation." It is difficult to fulfill both of these criteria, and empirical evidence is missing on how these criteria affect comprehension of the variability models. For instance, Fig. 1(b), which follows the constrained modeling style, contributes to the maximality property by addition of abstract choices "used to structure a feature model that, however, do not have any impact at implementation level" [72, p. 191]. The choices "Fuel," "Gear," "Drive," and "Gadget Level" are examples of abstract choices in Fig. 1(b) that increase the number of elements in the graphical model. The figure also fulfills the decomposition principle of minimality, as choices are not repeated in the graphical model. Classification focus is put on categorizing single choices into real-world-classes (e.g., classifying diesel and benzin as fuel). However, overall, the minimality property is violated because choices are redundantly mentioned in the textual constraints to specify the allowed configurations. In the hierarchical modeling style (see Fig. 1(a)), on the other hand, the decomposition principle of minimality is violated for the benefit of structural overview of choices, as choices are duplicated in the graphical model to implicitly express constraints. The choice "manual," for example, appears three times to constrain active-diesel, active-benzin, and adventure-diesel cars. Classification focus in the hierarchical modeling style is on representing the local choices at each node in the hierarchical structure.

In this context, it is interesting that specifying choices in variability modeling is not only based on logical structuring, but also on taking "additional ordering and grouping information" [19, p. 27] into account.

2.3. Cognitive effectiveness of variability modeling styles

We can now turn to a discussion on possible cognitive effects of different modeling styles as the hierarchical vs. constrained modeling styles.

From a cognitive point of view, working memory is the relevant brain system involved in inferring correct configurations from a model [4], and it is a limited resource. The cognitive load theory [69] describes how the design of information presentation affects cognitive load in working memory. Maximum capacity should be available for *germane cognitive load* – the processing of the information and the construction of schemas based on the information.

Intrinsic cognitive load is concerned with "the natural complexity of information that must be understood." [70, p. 124]. Complexity is primarily influenced by high element interactivity, namely, "elements that heavily interact and so cannot be learned in isolation" [70, p. 124]. To compare intrinsic cognitive load in our study, we define a dependence index, which aims to measure the degree of interaction between elements in a variability model: the higher the dependence index is, the higher the interaction between choices is ("choice interdependency"). The dependence index is not influenced by the modeling style and can be calculated for a specific problem domain that is characterized by the choices and dependencies to be modeled. The exact way to calculate the dependence index is described in Section 4, and noteworthy is that the dependence index of the situation modeled in Fig. 1 is relatively high irrespectively of the chosen modeling style, as "gadget level" implies constraints on "fuel" and "drive", whereas "gadget level" and "fuel" imply constraints on "gear" and "drive", and so on.

While it is not possible to change *intrinsic* cognitive load without changing the choices and their dependencies, the presentation of the variability models – e.g., the modeling style – can be altered, which might impose additional *extraneous* cognitive load. Extraneous cognitive load is influenced by the way information is represented [40]. In the context of our study, two cognitive load effects [70] dependent on extraneous cognitive load are relevant: the split-attention effect and the element-interactivity effect.

The *split-attention effect* [11] occurs when users have to not only split their attention between different sources of information but also to mentally integrate this information based on search-and-match processes, e.g., when text and diagrams are arranged spatially separated instead of in an integrated presentation [37,46]. Such a split-attention effect might occur in the case of combining a model with textual constraints, as is the case in the constrained modeling style shown in Fig. 1(b). As textual constraints often relate to more than one element in the model, there are no appropriate means to directly position them in the model.

Regarding *element-interactivity effect,* Sweller [70, p. 134] states that if "element interactivity due to intrinsic cognitive load is high, reducing the element interactivity due to extraneous cognitive load may be critical." We argue that if an element is repeated in a model, then the user is confronted with higher element interactivity, as more relations of the element to other elements have to be considered. The element-interactivity effect gives a clearer account of how to explain possible effects of different modeling styles than does the non-redundancy criteria of Czarnecki and Wasowski [19] and Parsons and Wand [52] because a repeated use of an element in a model may also serve for the correct model definition and does not represent redundancy of information. In these cases, repetition should not be considered 'unnecessary information' that could be eliminated. Repeating elements in a model also heightens the amount of model elements per se and will therefore heighten cognitive load, as users need to pay attention to a higher number of elements at the same time [40].

2.4. Variability spaces and CVL

Two variability spaces are commonly distinguished in the literature: problem space and solution space. The *problem space* deals with user goals and objectives, required quality attributes, and product usage contexts, whereas the *solution space* focuses on later development stages and refers to the functional dimension (i.e., capabilities and services), the operating environmental dimension (e.g., operating systems and platform software), and the design dimension (e.g., domain technologies) [39]. Traceability between those spaces is discussed in [6], where a conceptual variability model that allows a 1-to-1 mapping of variability between the problem space and the solution space is defined.

Referring to both problem and solution spaces, CVL facilitates the specification and resolution of variability over any base model defined by a metamodel. Its architecture consists of variability abstraction and variability realization. *Variability abstraction* supports modeling and resolving variability without referring to the exact nature of the variability with respect to the base model (the problem space). *Variability realization*, on the other hand, supports modifying the base model during the process of transforming the base model into a product model (the solution space).

In this study we concentrate on the variability abstraction part of CVL, which corresponds closely to feature models. The main examined concepts in our study are choices, their relationships, and constraints. *Choices* are technically similar to features in feature modeling. CVL offers more concepts for variability modeling, but our study does not apply them. Choice children are related to their parents higher in the tree in two different ways: (1) *Mandatory or optional*: the positive resolution of a child may be determined by the resolution of the parent (mandatory) or can be independently determined (optional). (2) *Group multiplicity*: a range is given to specify how many total positive resolutions must be found among the children: XOR/alternative – exactly one, OR – at least one.

Constraints express dependencies between choices of the variability model that go beyond what is captured by the tree structure. Two kinds of constraints are applied in our study: (1) A implies B – if A is selected, then B should be selected too (this constraint is known as "requires" in feature modeling), and (2) Not (A and B) – if A is selected, then B should not be selected and vice versa (this constraint is known as "excludes" in feature modeling).

3. Research model and hypotheses

Our goal is to examine whether the way variability models in general and CVL models in particular are organized influences comprehensibility and how. To this end, we refer to the two aforementioned modeling styles: *hierarchical*, in which most constraints are encoded in the tree hierarchy of the model, and *constrained*, which promotes a repetition-free visual classification tree, while cross dependencies are specified by textual constraints to restrict the possible set of configurations. We examine the ease of interpreting (reading) and creating (writing) the models mainly in terms of errors done and time to complete the task, but also by subjective means.

We summarize our expectations about the effect of modeling styles in two research frameworks: one for model interpretation (Fig. 2) and one for model creation (Fig. 3). In addition to the modeling style, we refer to the choice interdependency through the dependence index. As noted, this index measures the degree of interaction between choices in a model and is independent of the modeling style.

The first research framework proposes that CVL model comprehension is a function of the modeling style (extraneous cognitive load) and the choice interdependency (intrinsic cognitive load) – the dependency (or independency) between the involved elements. Highly dependent choices cannot be understood in isolation, and readers have to take all their relations with other choices into account. The first research framework further specifies that comprehension is measured both objectively (using the total score of correct answers and the time to complete the task) and subjectively (using users' scores for difficulty and ease of use).

In light of the theoretical considerations explained above, we will draw several propositions to investigate the effects of using different modeling styles on model readers' ability to comprehend the CVL model. Specifically, we build on cognitive load theory to explain possible effects of modeling style. We expect similar effects on objective as well as subjective model comprehension measurements and therefore the hypotheses are formulated for both.

As outlined above, separating textual constraints from the graphical model in the constrained modeling style might result in a split-attention effect for users. The split-attention effect heightens cognitive load and therefore, comprehension performance is expected to be lowered. However, the hierarchical modeling style may also lead to increases in cognitive load based on the element-interactivity effect because it might be necessary to integrate information from different occurrences of one and the same choice. Based on theory, we cannot determine which effect will be stronger. Thus, we want to investigate the hypothesis that:

H1. The modeling style influences comprehension of variability models.

Second, we want to discuss in which cases the split-attention effect caused by the constrained modeling style might be weaker than the element-interactivity effect of the hierarchical modeling style and vice versa. We argue that different levels of choice interdependency suit different modeling styles because depending on the situation, one cognitive load effect may be stronger than another. While the negative impact of high dependency on comprehension is obvious, we are interested in examining the interaction of the choice interdependency and the modeling style. Using the constrained modeling style for high dependency, for instance, can result in a high number of textual constraints (e.g., Fig. 1(b)) and thus in a higher number of repetitions leading to higher element interactivity. Such a case might be presented more efficiently with a hierarchical modeling style (see Fig. 1(a)) with a lower number of repetitions. For low dependency, it may be the other way around. Therefore, we propose:

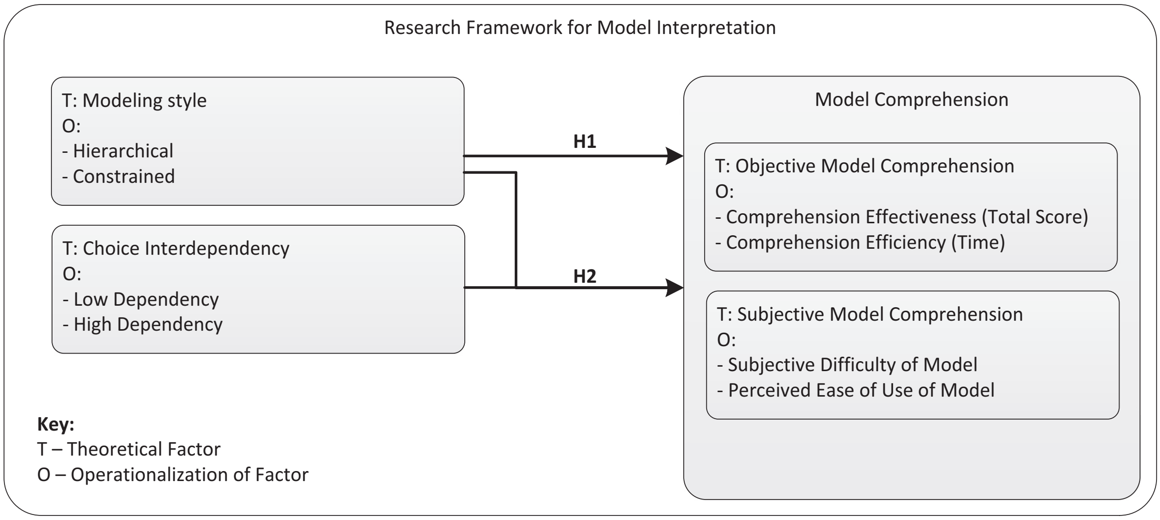

Fig. 2. Research framework for model interpretation.

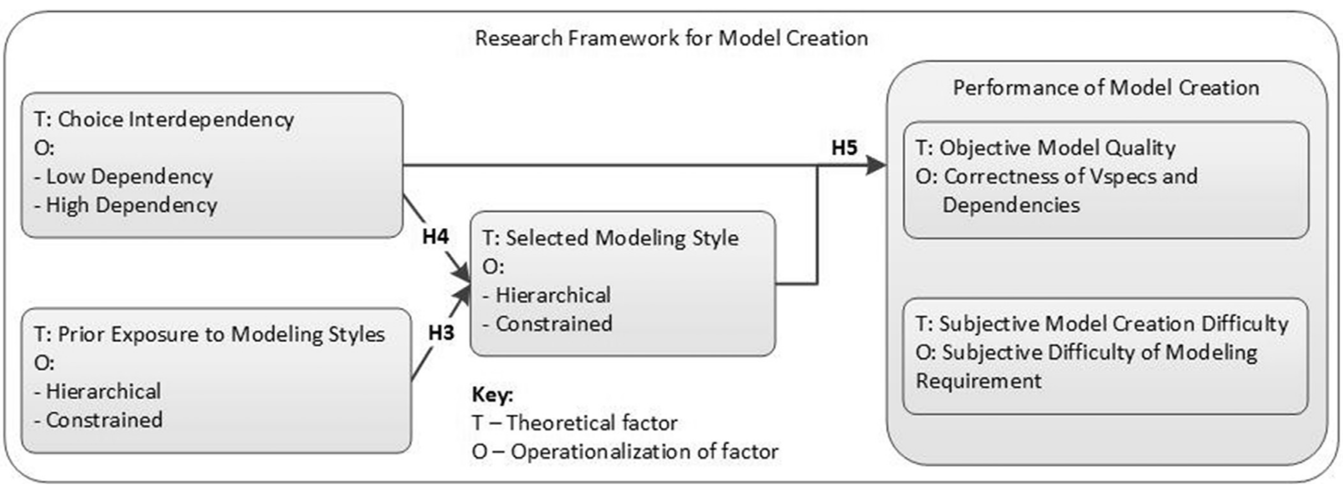

Fig. 3. Research framework for model creation.

H2. There is an interaction effect between the choice interdependency (as specified by the dependence index) and the modeling style, influencing model comprehension.

For discussing the expected effects of modeling style on model creation, we developed the research framework depicted in Fig. 3. This framework suggests that the choice interdependency (as specified by the dependence index) and prior exposure to a modeling style will influence the selection of the modeling style when creating models. The choice of a modeling style and the choice interdependency will result in differences in performance. Performance is measured in terms of effectiveness (i.e., correctness of choices, dependencies, and overall) and subjective difficulty reported by the users (on every requirement).

We first turn to the effect of prior exposure to modeling styles on selecting a modeling style for model creation. Use of examples prior to a design task can lead to a "functional fixation." Functional fixation is a cognitive bias to use an object the way it is normally used. Duncker defines functional fixedness as a "mental block against using an object in a new way that is required to solve a problem" [20] (e.g., using a hammer for pounding nails). Jansson and Smith [36] found that designers also tend to conform to examples provided to them in a conceptual design task. In light of these results, we hypothesize that modelers will also adhere to the modeling style exposed to previously:

H3. Prior exposure to a modeling style in examples leads to a higher subsequent use of this modeling style.

Second, we expect the choice interdependency to influence the selection of the modeling style. Modelers perceive that variability problems are of different kinds and this materializes through how constrained or hierarchical they make the description. For instance, if dependence is high (a high value on dependence index), it might be possible to model the case with a hierarchical modeling style using XOR relations similar to a decision tree structure, in which each path from the root to the leaf represents a valid choice configuration (see, e.g., Fig. 1(a)). In contrast, the constrained modeling style would need a variety of crosscutting constraints to represent the case correctly. On the other hand, for low dependency, it might be best to use various OR relations and only a few crosscutting constraints. Such a case is easier to define in a constrained modeling style – giving the whole combination possibilities first and then excluding single combinations. In this case, it might seem more difficult for participants to define all needed combinations to cover the whole configuration space in a hierarchical way. Thus, we hypothesize:

H4. The choice interdependency (as specified by the dependence index) influences the choice of modeling style.

Finally, we advance a hypothesis on how chosen modeling style might affect performance. We argue that the choice interdependency may call for a specific modeling style, and that applying the appropriate style implies higher performance (namely, higher quality of the created models and lower subjective difficulty). Accordingly:

H5. There is an interaction effect between the choice interdependency (as specified by the dependence index) and the modeling style, influencing the quality of the created models and the perceived difficulty.

4. Experimental design and procedure

4.1. Experimental design

To test our hypotheses, we used a between-groups design with one main factor (modeling style) with two levels (constrained vs. hierarchical). In each experimental group, participants were shown two models describing the variability within different sets of Skoda Yeti choices (basic choices as depicted in Fig. 1 and extra choices – in Appendix A), both modeled either in the constrained or in the hierarchical style. As explained later, the basic and the extra models differ in terms of dependency between choices. The participants were asked first to answer comprehension questions about these two models without using any supporting tool. We varied the order of the two models to control for possible learning effects. Next, the participants had to create a CVL model (using a dedicated CVL tool, as explained later) themselves based on a short natural language description of the choices in the top-of-the-range Skoda Yeti edition without being guided regarding the modeling style.

4.2. Materials and measurement of variables

We used an online questionnaire with four parts: pre-questionnaire, studying, comprehension part, and modeling part. We next elaborate on each part.

4.2.1. Pre-questionnaire

The purpose of the pre-questionnaire was to obtain general information about the participants and their background, including age, gender, degree and subject of studies, and familiarity with feature modeling (this was the only variability modeling approach the participants could be exposed to). The familiarity of participants with variability modeling is important, as experts develop 'schemas' – language-independent, abstract problem representations – in their mind, e.g., for programing [60] or modeling constructs [25]. They therefore have more working memory resources available for comprehending the model. To measure (self-rated) familiarity with feature modeling, we adopted the three-item modeling grammar familiarity scale of Recker [54] with a 7-point Likert scale (from strongly disagree to strongly agree): (1) overall, I am very familiar with feature diagrams, (2) I feel very confident in understanding feature diagrams, and (3) I feel very competent in modeling feature diagrams.

4.2.2. Studying

After filling in the pre-questionnaire, the participants were presented with slides explaining and exemplifying the relevant parts of CVL. The participants were also given hard-copies of these slides, which they could consult while answering the questions. The participants had to study CVL on their own from the slides and proceed to the main questionnaire.

4.2.3. Comprehension part

4.2.3.1. The models. In the comprehension part, each participant received two CVL models following the same modeling style describing different sets of choices of Skoda Yeti cars and their variability. One model describes basic choices, such as fuel and drive (see Fig. 1), and the other describes extra choices, such as panorama roof and parking heater (see Fig. 5 in Appendix A). Although the numbers of choices in those models are quite similar, the choice interdependency differs. To calculate the degree of dependency in each modeling situation, we calculate dependence indices as follows. For each pair of distinct non-abstract choices (apart from mandatory and dead choices[2]), we count the number of combinations allowed in all valid configurations generated from the given model. The maximal number is 4 – Ø, {A}, {B}, {A, B} – to represent no selection of the two choices, the selection of A, the selection of B, and the selection of both A and B, respectively. Any dependency reduces one or more combinations. The dependence index is then calculated as 1 minus the normalized sum of the above numbers for all pairs (where each pair is considered once, irrespective of the order of choices).[3] Dependence index of 0 means that all choices are independent of each other, i.e., all the four combinations are feasible for each pair of choices. The upper bound of the dependence index is .5 since our process will never result in less than two combinations for each pair of choices.[4] The closer the dependence index is to 0, the less dependency between choices exists. Note that the dependence index is not influenced by the modeling style, as superfluous choices, e.g., resulting from classification, are not included in the calculation. The dependence index of the models in Fig. 1 ("basic") is .19 (see Appendix B for the calculation details), indicating relatively high dependency between choices. The dependence index of the models in Fig. 5 in Appendix A ("extra"), on the other hand, is clearly lower – .05 – indicating low choice dependency.

Overall, we had two experimental groups: one in which the two models were specified following the hierarchical modeling style and the other in which the two models were specified following the constrained modeling style. However, we had four questionnaire variants (as indicated in Table 1) because we also varied the order of the two models in each of the two experimental groups to control for possible learning effects.

The CVL models for the experiment were built by Haugen, one of the creators of CVL who is familiar with the possible Skoda Yeti configurations from the Norwegian Skoda public web pages. All authors checked that the versions of the same model ("basic," "extra") fulfill the requirement of informational equivalence, meaning that "all information in one [representation] is also inferable from the other and vice versa" [43, p. 67]. In the context of variability modeling, each pair of models can be described as "equivalent" because their configuration space is equal, namely, "the set of all instance descriptions derivable from the first diagram is equal to the set of instance descriptions derivable from the other diagram" [15, p. 86].

[2] Mandatory choices appear in all valid configurations and hence should not have a contribution to dependency calculation. Dead choices do not appear in any configuration and are thus redundant in the specification. As such, they should not be taken into consideration in the calculation of the dependence index.

[3] The normalized sum is achieved by dividing the sum by the maximal potential one, i.e., $4 \times n \times (n-1)/2$.

[4] It is obvious that no pair of choices can give 0 combinations, since every configuration will give one pair of truth values. If the number of truth values is 1, then this means that the two choices in question are constant over the set of configurations, but such situations – mandatory choices – have been eliminated by our process.

Table 1
Questionnaire variants.

Experimental group	Variant number	First model	Second model
A	1	Basic, hierarchical	Extra, hierarchical
	2	Extra, hierarchical	Basic, hierarchical
B	3	Basic, constrained	Extra, constrained
	4	Extra, constrained	Basic, constrained

4.2.3.2. Comprehension tasks. For both experimental groups, ten questions were asked about each model ("basic," "extra"), examining whether specific configurations of Skoda Yeti cars are allowed (see Appendix C for the full list of questions). These questions can be described as surface-level tasks (measuring comprehension of models more directly than do deep-level tasks), which require participants to work with the models in a usage context [50]. Moreover, as CVL models aim at representing variability, comprehending which configurations are valid and which are not is the main task for investigation.

The participants were presented with the model, one question at a time. They had to choose between the following answers: *correct*, *wrong*, *cannot be answered from model, I do not know*. After answering a question, participants proceeded to the next question, but could not return to previous questions. No rigid time constraints were imposed on the participants.

As noted in Section 3, we measured the cognitive effect of the modeling styles on comprehension using two objective measures: comprehension effectiveness – operationalized with correctly answered questions on model content – and comprehension efficiency – time needed to answer the set of questions regarding the model content. Such measures of effectiveness and efficiency are widely used in investigating comprehension of conceptual models [33].

4.2.3.3. Post comprehension questionnaire. The participants had to fill a post-part questionnaire that collected subjective ratings of the comprehension of each model. In particular, we measured perceived ease of use of the model with a slightly adapted version of the 4-item scale of [45]. An example item was "Learning how to read the model was easy." We further measured the difficulty in understanding different model constructs relevant to variability modeling (mandatory and optional elements, XOR and OR relations), with the answering options ranging from 1 = very easy to 7 = very difficult. In addition, the participants could report on difficulties they experienced in open text fields.

4.2.4. Modeling part

After completing the comprehension task, the participants were given a short tutorial of a CVL tool including operations such as adding choices, setting groups, and defining constraints. The tool was an early stand-alone version of what has now become the BVR Tool.[5] The participants would get immediate help if they had tool problems, but this was extremely rare as the tool itself was easy to grasp for our modeling task. In addition, they got a short textual description (two paragraphs) of a top-of-the-range edition of Skoda, called Laurin and Klement. The modeling task focused on this top-of-the-range edition and on its diesel cars (see Appendix D). The choices were quite obvious within the description, so that we will be able to concentrate on the organization of the choices into diagrams (in the form of modeling styles) rather than on their extraction from the text. Although often variability models are automatically created from software development artifacts (for example, from requirements [35]), the aim of this task was to check the difficulties humans face when specifying variability models, e.g., in scenarios of modifying automatically-generated models due to changes in the variability requirements.

The participants were given hard copies of the tutorial and the description, and they were free to consult the hard copies when creating the model. The only requirement was to apply the given tool in order to prevent syntax errors. The constraints could be given either as a parsed text in the tool or as a free text separate from the tool. We ignored "simple" syntax errors when analyzing the constraints.

Similar to the comprehension part, the modeling task also referred to basic and extra choices, although in a slightly different way from those of the comprehension part. The dependence index of the "basic" model was higher than that of its counterpart in the "extra" model (.3 and .04, respectively).

After completing the modeling task, the participants were asked to rate the difficulty of each requirement they were requested to model. The rates ranged from 1-very easy to 7-very difficult.

We measured the quality of model creation in terms of correctness, as well as the reported difficulty to do that. As the participants were free to choose any modeling style, we observed mixed modeling styles, in addition to the pure ones – constrained and hierarchical. The way we chose to handle these cases is elaborated upon later.

4.3. Sample

Participants were recruited from four different classes (in three different countries) from information systems, informatics, and business curricula with prior training in modeling. In each class, the participants were arbitrarily divided into the four combinations of experimental groups and experimental orders (see Table 1). To assure sufficient motivation during the experiment, participants received approximately 5% course credit for this task, but they could decide not to participate in the experiment at all, as this credit was either defined as a bonus or could be substituted by another task, depending on the class. Nevertheless, most students chose to participate in the experiment.

We performed a power analysis using the G*Power 3 software [22] to approximate sample size requirements for a subsequent ANCOVA (analysis of covariance) with one covariate across two groups (modeling style) and expecting medium effect sizes of $f(U) > .30$ with type-1 error probability of $\alpha \langle .05$. A sample size of $n = 90$ was required to reach sufficient statistical power $(\rangle .80)$.

A total of 92 students participated in the study, thus fulfilling the sample size criterion. Examining their background, we found that the number of models previously created or read was negatively skewed – there were a few very experienced modelers and mostly plain novices. We decided to exclude univariate outliers based on the criterion "standardized scores in excess of 3.29″ [71, p. 73]. Therefore, two participants (who had created or read over 200 models) were excluded, reducing the sample size to 90 (43 and 47 participants per experimental group, respectively). Table 2 gives relevant demographic statistics for both experimental groups. We performed t-tests and X^2 tests to screen for possible differences between the experimental groups. Results did not suggest significant differences between groups.

[5] http://modelbased.net/tools/bvr-tool/.

Table 2
Participants' demographic data (M = mean, SD = standard deviation).

	Hierarchical ($n = 43$)		*Constrained* ($n = 47$)		Total ($N = 90$)		Statistical test
	M/count	SD/percentage	M/count	SD/percentage	M/count	SD/percentage	
Age	25.42	3.16	25.31	4.38			$T_{df=86} = 0.13$; *n.s.*
Gender							
Female	18	42%	20	43%	38	42%	$X^2_{df=1} = 0.004$; *n.s.*
Male	25	58%	27	57%	52	58%	
Amount of models created or read	31.61	36.86	25.64	35.43			$T_{df=88} = 0.78$; *n.s.*
Work experience as programer							
Yes	7	16%	12	25%	19	21%	$X^2_{df=1} = 1.15$; *n.s.*
No	36	84%	35	75%	71	79%	
Familiarity with software product line engineering							
Yes	17	40%	14	30%	31	34%	$X^2_{df=3} = 0.95$; *n.s.*
No	26	60%	33	70%	59	66%	
Familiarity with feature modeling (3 items, mean value, 7-point scale, from 1 = strongly disagree, 7 = strongly agree)	2.67	2.22	2.45	1.79	2.55	2.00	$T_{df=80.78} = 0.51$; *n.s*

We further analyzed for differences between the courses where our study took place. As those analyses are not related to the main research questions, we present the results in Appendix E.

5. Results

Data analysis was carried out with SPSS 20.0. We elaborate next how data were analyzed and then detail the results.

5.1. Model comprehension

5.1.1. Data screening

To test our first research framework, including hypotheses 1 and 2, we ran four repeated measure analyses of covariance (ANCOVAs) with *experimental group (constrained* vs. *hierarchical) and experimental order (first* or *second* task) as between-subject variables. The dependent variables in the four separate ANCOVAs were *comprehension effectiveness (correctness), comprehension efficiency (time), subjective difficulty of model,* and *perceived ease of use*. Each dependent variable was measured twice (for each of the two models: high choice interdependency (*basic)* and low choice interdependency (*extra*)), thus constituting a within-subjects factor. *Familiarity with feature modeling* was used as model covariate (a controlled variable).

In a first step, we checked whether assumptions for performing ANCOVAs for repeated measures were met based on the procedures proposed in [71]. Shapiro–Wilk tests of the dependent variables indicated that the assumption of normality of dependent variables had been violated. However, ANCOVAS' robustness is expected with at least 25 participants per experimental condition [63] and we had over 40 participants per experimental group.

We sought univariate outliers within each experimental group because they might distort statistical analyses [71, p. 72]. Concerning the model with high choice interdependency – basic – we had to exclude two univariate outliers in the analyses. Based on the criterion "standardized scores in excess of 3.29″ [1, p. 73], we excluded two cases for all four analyses (one out of each experimental group) because these participants had used a high amount of time for solving questions on the basic model (906 and 812 s). It is possible that these subjects were distracted during the experiment.

Box's M tests for homogeneity of variance-covariance matrices indicated potential problems with homogeneity of variance for all four analyses. Therefore, we assessed homogeneity of variance with $F_{\max}$ (ratio of largest to smallest cell variance) [71, p. 86]. Since our sample sizes were relatively equal and $F_{\max}$ was lower than 5 in the analyses, we deem this assumption to be met.

5.1.2. Tests of hypotheses

Table 3 and Fig. 4 give an overview of the results of the ANCOVAs for repeated measures. Overall, there was a significant effect of the experimental group (constrained vs. hierarchical) on all dependent variables. The modeling style did influence comprehension effectiveness, lending support to H1. The hierarchical modeling style was easier to comprehend. However, there is also a significant disordinal (crossover) interaction effect of choice interdependency and modeling style, indicating that the effect of modeling style differs for the "basic" and "extra" models. This means that the type of effect the modeling style has depends on the choice interdependency of the models, thus supporting H2 predicting an interaction effect. Therefore, the main effect of modeling style cannot be interpreted without taking the choice interdependency into account. While participants achieved a higher comprehension of the model with high dependency in the hierarchical test condition ($F(1,83) = 25.07$, $p < .001$), they significantly understood the model with low dependency better in the constrained style ($F(1,83) = 4.26$, $p = .04$). Similarly, participants took less time for answering questions for the hierarchical model with high dependency ($F(1,83) = 23.69$, $p <= .001$) than for the constrained model with high dependency, and took less time for answering questions for the constrained model with low dependency ($F(1,85) = 8.34$, $p = .005$) than for the hierarchical model with the low dependency. These results provide evidence to accept H2 in terms of objective comprehension.

Next, we discuss results of the subjective model comprehension. Participants rated the ease of use of the hierarchical model higher than that of the constrained model. Additionally, they rated the subjective difficulty of the hierarchical model lower for both models. These results support H1 concerning the effect of modeling style on subjective comprehension of CVL models. While there was an interaction effect of choice interdependency and modeling style for perceived ease of use, there was no interaction effect for the subjective difficulty of the model, and thus the subjective data did not provide clear support for H2. Concerning ease of use, participants perceived hierarchical models easier to use in both models, where in the model with high dependency – basic – the differences were even larger with respect to the constrained model.

Table 3
An overview of the results of the ANCOVAs for repeated measures.

	Effect	F ($df_{\text{Hypothesis}} = 84$; $df_{\text{Error}} = 1$)	Significance	Partial eta squared
Comprehension effectiveness (total score)	*Modeling style*	4.21	.04	.05
	Choice interdependency		n.s.	
	*Choice interdependency * modeling style*	36.51	<.001	.30
	Experimental order		n.s.	
	Familiarity with feature modeling	7.50	.01	.08
	*Choice interdependency * experimental order*	4.57	.04	.05
	*Choice interdependency * familiarity with feature modeling*		n.s.	
Comprehension efficiency (time)	*Modeling style*		n.s.	
	Choice interdependency	6.64	.01	.07
	*Choice interdependency * modeling style*	43.81	<.001	.34
	Experimental order		n.s.	
	Familiarity with feature modeling	12.08	.001	.13
	*Choice interdependency * experimental order*	34.17	<.001	.29
	*Choice interdependency * familiarity with feature modeling*		n.s.	
Perceived ease of use	*Modeling style*	24.80	<.001	.23
	Choice interdependency		n.s.	
	*Choice interdependency * modeling style*	9.21	.003	.10
	Experimental order		n.s.	
	Familiarity with feature modeling	4.92	.03	.06
	*Choice interdependency * experimental order*		n.s.	
	*Choice interdependency * familiarity with feature modeling*		n.s.	
Subjective difficulty of model	*Modeling style*	4.14	.05	.05
	Choice interdependency		n.s.	
	*Choice interdependency * modeling style*		n.s.	
	Experimental order		n.s.	
	Familiarity with feature modeling	17.07	<.001	.17
	*Choice interdependency * experimental order*		n.s.	
	*Choice interdependency * familiarity with feature modeling*		n.s.	

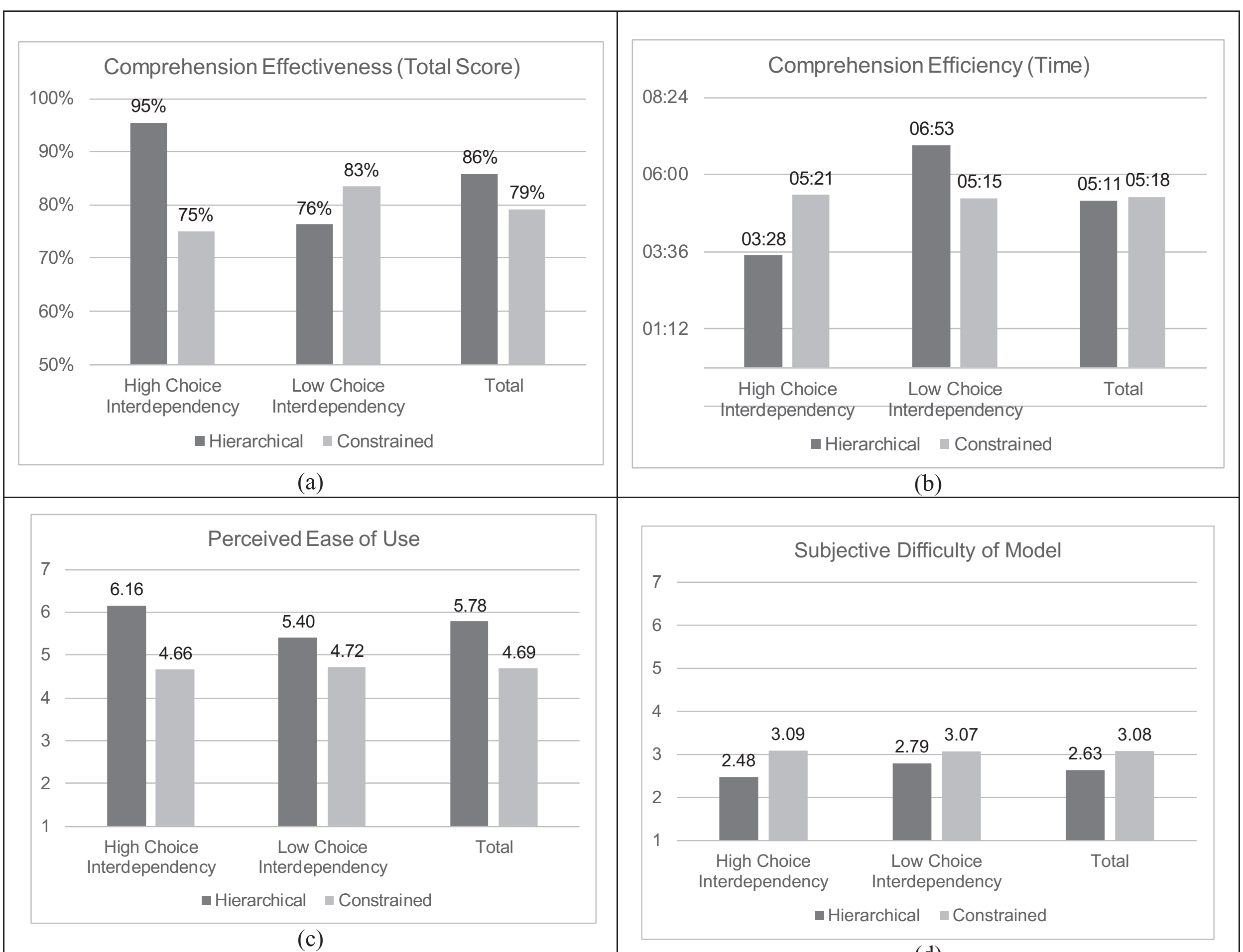

Fig. 4. Results for model comprehension: (a) comprehension effectiveness (total score), (b) comprehension efficiency (time), (c) perceived ease of use, and (d) subjective difficulty of model.

Table 4
Examples of decomposing requirements into examined elements.

Requirement	Examined element	Element type
When it is automatic, only the 4 × 4 drive and a 140 hp engine are possible.	4 × 4	Choice
	140 hp	Choice
	Automatic -> (4 × 4 and 140 hp)	Dependency
Choosing the parking assistant excludes choosing the backing sensor.	Parking assistant	Choice
	Backing sensor	Choice
	Parking assistant -> not (backing sensor)	Dependency

As for the controlled variables, the experimental order did not influence comprehension directly. However, there were significant interaction effects between the choice interdependency and the experimental order for comprehension effectiveness (total score) and efficiency (time). While comprehension scores in the basic model did not depend on the order, the extra model was better understood when being second than when being first. Participants did answer more questions (83% vs. 77%) on the extra model correctly in cases where they had previously worked on the basic model. They did use more time on the first (basic = 328.72; extra = 409.09 s), compared to the second model (basic = 235.50; extra = 313.67 s) on which they were answering questions, regardless of which model was first – basic or extra. There was no effect of experimental order on perceived ease of use nor on subjective difficulty of model.

Familiarity with feature modeling did have an effect on all dependent variables, which is in line with prior studies on comprehending variability models by novices and experts [57]. Participants with higher familiarity performed better on the comprehension tasks, but they also took more time to solve them. They rated the perceived ease of use of the model higher and the difficulty to understand different model elements as lower, with respect to participants with lower familiarity.

5.2. Model construction

5.2.1. Data screening and coding

The number of different models created for the given natural language description was quite large. Therefore, two of the authors of this paper encoded the created models independently. Two models belonging to the hierarchical modeling group were missing because the participants failed to upload the correct files for their solution. Overall, 88 models were analyzed. For each model, the specification of each requirement (a sentence or a part of a sentence in the textual description) was separately encoded. Moreover, each requirement was decomposed into choices and dependencies among them. Table 4 provides some examples of this decomposition.

The specification of each element (choice or dependency) could be completely correct, partially correct, incorrect, or missing (i.e., no evidence that the participant noticed the requirements for the element). We assigned 1 point for each correct answer and .5 points for a partially correct answer. At this stage, we have not differentiated between missing and incorrect specifications. The encoders further examined the dependencies among choices as crosscut (textual) constraints or hierarchical dependencies (including OR, XOR, optional, and mandatory relations); accordingly, the representation type could be text or model, respectively. The typical case was that a single dependency was specified either as a single textual constraint or in the model. There were a few cases in which a constraint was modeled as part of another constraint (three cases) or as two distinct textual constraints (three cases). These six cases were also treated as 'text' during the entire procedure.

The encoders further classified the modeling styles used to specify the basic and extra choices of Laurin and Klement cars (hierarchical vs. constrained).

After independently encoding all models, the encoders discussed the differences in their coding until they reached full agreement.

5.2.2. Tests of hypotheses

First, we turn to the effect of prior exposure on choice of modeling styles (our third hypothesis – H3). To compare experimental groups, we used chi square tests (see Table 5 for descriptive results and all test values). From the data in Table 5, it is apparent that there is a significant influence of prior exposure to modeling style on the style chosen, both for the basic model ($X^2_{df=1} = 33.76$; $p < .001$) and the extra model ($X^2_{df=1} = 22.57$; $p < .001$), the effects of which can be considered large (Phi $\varphi = .62$ and .51, respectively). In 76% of the cases, the participants stuck to the modeling styles to which they were exposed. If beforehand confronted with constrained models in the first part of the experiment (the comprehension task), participants used a constrained modeling style more often than using a hierarchical style (83% vs. 17%, overall). When confronted with a hierarchical style, participants stuck to the hierarchical style in 68% of the cases, while in 32% of the cases they switched the modeling style to a constrained style. Overall, the results lend support to hypothesis H3, that prior exposure affects choice of modeling style.

From Table 5 we can derive that it depends on the choice interdependency whether hierarchical or constrained modeling styles are chosen: more (60%) participants chose to model the high dependency model in a hierarchical style than they did in a constrained style (40%) and vice versa for the low dependency model (22% hierarchical style vs. 78% constrained style). Thus, the results support H4, that choice interdependency may further influence whether users choose a specific modeling style (see Table 6).

Next, we turn to the effects of the chosen modeling style and the choice interdependency on the resulting model (our fifth hypothesis – H5). To this end, we calculated *t*-tests for independent samples for the basic model and for the extra model. Cohen's d was calculated in a separate tool[6] to determine effect sizes for significant effects. Results show a positive overall influence of the constrained modeling style on modeling correctness, both for the model with high choice dependency (Constrained: M = .89, SD = .17; Hierarchical: M = .82, SD = .16; $t_{df=86} = -1.82$, $p = .07$) and the model with low choice dependency (Constrained: M = .89, SD = .09; Hierarchical: M = .80, SD = .17; $t_{df=21} = -2.26$, $p = .04$). When looking at a detailed level, we note that the constrained modeling style had only a positive influence on choices in the model with low choice dependency, but not in the model with high choice dependency. The absence of a measurable effect for choices in the high dependency model might be due to the fact that most choices in this situation were involved in several dependencies and hence were unavoidable.

[6] Ref: http://wilderdom.com/301/Cohensd.xls, last retrieved 20/04/2016.

Table 5
Prior exposure and choice of modeling style.

	Prior exposure: hierarchical style ($n=41$)		*Prior exposure: constrained style* ($n=47$)		*Average percentage* corrected with group size of prior exposure)	Statistical test		
	Count	*%*	*Count*	*%*	*%*	$X^2_{df=1}$	*P*	φ
Model with high choice dependency – basic choices								
Hierarchical	38	93	15	32	60	*33.76*	*<.001*	.62
Constrained	3	7	32	68	40			
Model with low choice dependency – extra choices								
Hierarchical	18	44	1	2	22	*22.57*	*<.001*	.51
Constrained	23	56	46	98	78			

Table 6
Choice of modeling style and correctness of models.

	Hierarchical modeling style		*Constrained modeling style*		Statistical test		
	M/count	SD/%	M/count	SD/%	$T_{df=86}$	*p*	*Cohen's d*
High choice dependency – basic choices	($n=53$)		($n=35$)				
Correctness							
Choices	.99	.04	.98	.11	*.37*	*.71*	–
Dependencies	.58	.38	.75	.31	*2.31*	*.02*	*−.49; small effect*
Overall	.82	.16	.89	.17	*1.82*	*.07*	*.43; small effect*
Subjective difficulty	2.85	1.10	2.61	1.40	*.89*	*.38*	–
Low choice dependency – extra choices	($n=19$)		($n=69$)				
Correctness							
Choices	.93	.09	.98	.06	*−2.24*	*.04*	*−.75; moderate effect*
Dependencies	.57	.34	.73	.22	*−1.92*	*.07*	*−.65; moderate effect*
Overall	.80	.17	.89	.09	*−2.26*	*.04*	*−.82; large effect*
Subjective difficulty	3.11	.99	2.72	1.51	*1.05*	*.30*	–

We further note that the constrained modeling style had a positive influence on correctness of dependencies in both situations (although the difference is statistically significant at the $\alpha = .05$ level only in the model with high choice dependency). Overall, we deem hypothesis H5 to be rejected: the use of a constrained modeling style results in higher model correctness for both models. Concerning subjective modeling difficulty, we did not find any significant difference – whether hierarchical or constrained modeling style was chosen – and thus H5 was not supported by subjective measures either.

6. Discussion

This study set out with the aim to examine hierarchical and constrained styles in variability modeling. A main finding of this study is that differences in comprehension and selection of a specific modeling style depend on choice interdependency. While for a high choice dependency situation, the hierarchical style was easier to understand and also chosen more often to create a model, for a low choice dependency situation the constrained version performed better in terms of comprehension effectiveness and efficiency and was also chosen more frequently to model. Table 7 summarizes the hypotheses testing results. In line with our predictions, these combinations of modeling style and choice interdependency led to a lower number of occurrences of the (non-abstract) choices in the models and thus a lower element-interactivity effect, which would have heightened cognitive load. This is also reflected in additional analyses based on comprehension question type (see Appendix E): question-based redundancy of choices was in general higher for the model with high choice dependency in the constrained style and for the model with low choice dependency in the hierarchical style. The constrained modeling style outperformed the hierarchical style for comprehension questions that lead to much lower redundancy in the constrained style, but not in case it leads to equal or higher redundancy. Thus, it seems that the effect of element-interactivity was more important than the effect of split-attention between textual constraints and the graphical model in the constrained modeling style. If the negative effect of the split-attention effect would have been very strong, both models should have been easier in the hierarchical style.

Our results show that the level of choice interdependency has an impact on what style should be applied in order to obtain the most comprehensible model. They further indicate that the selection of the modeling style depends on the degree of dependency. There seems to be a common understanding of modelers as to when to use the different modeling styles, which can be seen by how modelers "naturally" chose different styles for different levels of choice interdependency (controlled for their tendency to choose the style they were exposed to earlier).

However, we found two exceptions from this overall pattern, which we discuss below. First, the hierarchical modeling style was subjectively rated to be easier in both models. Second, we did not find that applying the appropriate modeling style to a specific choice interdependency situation would result in better model quality in any of the two models, as models in the constrained modeling style had fewer errors.

Regarding the subjective model comprehension of the hierarchical modeling style, participants interestingly rated it higher for both models. Prior research has demonstrated that preference for a representation format might not always correspond to performance in using the representation [14]. While objective comprehension values were lower for the hierarchical model in the extra task (low choice interdependency), users still rated the comprehensibility higher. This result is in line with hypothesis 1,

Table 7
Summary of hypothesis testing results.

Hypothesis	Dependent variable	Results
H1. The modeling style influences comprehension of variability models.	H1a. Comprehension effectiveness	Supported. The constrained modeling style leads to less comprehensible models. There is a significant disordinal (crossover) interaction effect of dependence index and modeling style.
	H1b. Comprehension efficiency	Not supported.
	H1c. Perceived ease of use	Supported. The constrained modeling style leads to lower subjective model comprehension.
	H1d. Subjective difficulty of model	Supported. The constrained modeling style leads to higher subjective difficulty.
H2. There is an interaction effect between the choice interdependency (as specified by the dependence index) and the modeling style, influencing model comprehension.	H2a. Comprehension effectiveness	Supported. Participants achieved a higher comprehension of the model with high dependency in the hierarchical style, while they understood the model with low dependency better in the constrained style.
	H2b. Comprehension efficiency	Supported. Participants took less time for answering questions for the high dependency model in the hierarchical style, and took less time for answering questions for the low dependency model in the constrained style.
	H2c. Perceived ease of use	Supported. The relative higher rating of perceived ease of use of the hierarchical model style is more prominent for the case of low dependency than for the case of high dependency.
	H2d. Subjective difficulty of model	Not supported.
H3. Prior exposure to a modeling style in examples leads to a higher subsequent use of this modeling style.		Partly supported. The effect is clear for the combinations of basic model (high choice interdependency) × hierarchical modeling style and extra model (low choice interdependency) × constrained modeling style; while in the other two combinations switches occur.
H4. The choice interdependency (as specified by the dependence index) influences the choice of modeling style.		Supported. Hierarchical style was chosen more often for the model with high choice dependency, the constrained style was chosen more often for the model with low choice dependency.
H5. There is an interaction effect between the choice interdependency (as specified by the dependence index) and the modeling style, influencing the quality of the created models and the perceived difficulty.	H5a. Model correctness	Not supported. The constrained modeling style results in higher quality models for both models.
	H5b. Subjective difficulty	Not supported.

that the modeling style affects comprehension: the results suggest that users perceive the split-attention effect between textual constraints and model more strongly than the element interactivity effect of repeated choices in the hierarchical modeling style; thus they rate comprehensibility lower for the constrained models.

There could be several different interpretations of the higher subjective comprehension of the hierarchical modeling style. The results could be interpreted in light of the "hidden dependencies" – users might have had the impression that there were more hidden dependencies based on combinations of constraints in the constrained model, while in the hierarchical model such dependencies could have been easier to recognize. Haisjackl et al. [30] report a similar effect in the area of declarative process models – that "hidden dependencies" based on combinations of constraints are a challenge for model comprehension. Another possible interpretation of the higher subjective comprehension of the hierarchical modeling style can be derived from the ontological literature. Textual constraints (especially those in the form "not (A and B)") presumably have a similarity to the ontological construct "negated property – a property a thing does *not* possess." [8, p. 387]. Bodart et al. [8] argue that humans do not easily perceive such properties. Thus, models expressed in the constrained modeling style (including such constraints) might be experienced as being more difficult than hierarchical models that visualize all possible options.

As to why modeling in the constrained modeling style leads to higher model quality independent of the choice interdependency, different arguments can be used, e.g., textual constraints can directly be taken from the natural language description or separating concerns in graphical and textual parts helps modelers to model correctly. In contrast to the comprehension of existing CVL models – creating constrained CVL models seems to be less error-prone than is creating hierarchical CVL models. The user can first create a redundancy-free hierarchical model of the choices and then add missing constraints as textual additions. The split-attention effect is less likely to happen if the task is performed in a sequential, rather than in a parallel, order, as in the comprehension task. The results may also be caused by a similarity of the constrained modeling style with other widespread visualizations that employ "redundancy-free" node-link diagrams, in which each concept is only mentioned once, e.g., concept maps [21].

We are aware that we cannot give a definite answer as to why the constrained modeling style proved to be more effective in terms of quality of the resulting models. In future investigations, we encourage the exploration of the "process of variability modeling," e.g., by tracking modeling steps by the editor and analyzing them as has been done in other modeling areas. Such data would help clarify why modeling in a constrained way seems to be more beneficial than comprehending models in a constrained modeling style [67].

Our results further indicate that for relatively inexperienced users, as in our sample, it is easier to get models right using the constrained style; nevertheless, the hierarchical style is easier to comprehend from a subjective point of view. We thus postulate that it may be worthwhile to put extra effort into making a hierarchical model, since it would be better understood in the sequel. It may also be the case that with more experience, variability modelers would be more inclined to use the hierarchical style.

Our results further indicate that the choice of the modeling style depends not only on the degree of dependency, but also on the prior exposure of the modelers to modeling styles. Visual example models may have a possible constraining effect and lead to inappropriate models, because modelers adhere to them. However, we observed that modelers did not blindly adhere to given examples, but adapted to the specific circumstances of the given choice interdependency. Half of the participants presented with hierarchical style models first, switched to the constrained modeling style for modeling a low dependency modeling situation. In general, prior exposure seems to be stronger for the constrained modeling style than for the hierarchical modeling style, as more participants stick to it. A possible interpretation may be that participants might sense in which style they make fewer errors and perform better. Switching to the constrained modeling style therefore seems to be a wise decision, as models modeled in a constrained style showed a higher correctness, especially for modeling dependencies, for both models with high/low choice interdependency.

7. Implications and threats to validity

7.1. Implications for research

In terms of research, the current findings add strength to a growing body of empirical work that supports the cognitive load theory in the conceptual modeling field. The fact that the appropriateness of a modeling style is highly dependent on the choice interdependency can also be seen as an extension of the cognitive fit theory, which postulates that cognitive fit between the task type and the information emphasized in the representation leads to more effective and efficient problem solving. Thus, even for one and the same task (as model comprehension tasks), different representations may be beneficial, depending on the inherent structure of the information to be represented. Of course, there are many more aspects of extrinsic cognitive load that the present study has not looked into. These range from presentation medium (paper versus computer), over primary notation (other notations rather than CVL), notational characteristics as semantic transparency and perceptual discriminability of symbols and secondary notation – related to aspects not formally defined – as the use of decomposition into sub-models, color highlighting or layout of the model and the labels. When modeling in a tool, also usability aspects are relevant. These general variables, relevant to any type of conceptual model, were held constant for experimental purposes to determine the effect of the variables that are of specific interest to variability modeling.

The study also took a look at whether a split-attention effect (between textual constraints and graphical modes) would be stronger than an element-interactivity effect (caused by redundantly modeled choices). In our experiment, the element-interactivity effect was stronger. However, caution must be applied when generalizing the result we obtained, because we used only two different models. Furthermore, it was not possible to compare comprehension questions according to their degree of split-attention effect, because all questions lead to a split-attention between model and text in the constrained modeling style. Therefore, to meaningfully examine the strength of split-attention effects in this context, we advise fellow scholars to systematically construct comprehension questions (similar to e.g. [25]) varying the existence and strength of a split-attention effect. Future research on cognitive load effects for conceptual models is advised. Turetken et al. [74] have for instance investigated such effects in decomposition of models and hierarchical structuring. They reported no evidence of increased comprehensibility from using abstraction (which would aid comprehension); on the contrary, tasks that required information from sub-processes were answered better when this information was not hidden (and, thus, no split-attention effect, which would lower comprehension, could occur). While the results cannot be compared to the present study, they also demonstrated that it would be important in the future to collect data on more modeling cases to be able to specify tradeoff curves between competing positive and negative cognitive load effects on comprehension.

Our study further shows a high conformance with prior model examples in terms of modeling style when creating a new model. This study thus extends research on fixation effects in design tasks, which have predominately been examined in architectural or mechanical design tasks [13,36,66] to the area of conceptual modeling.

While choosing a constrained modeling style leads to higher quality of resulting models, it was somewhat surprising that models in the constrained modeling style were judged to appear less comprehensible. This finding suggests that results of model comprehension tasks cannot necessarily be transferred to model creation tasks and vice versa, and researchers have to exercise caution when generalizing results in cases where only one task type (model comprehension vs. model creation) is considered.

7.2. Implications for practice

The study presented in this paper has implications for modeling practice and is of direct practical relevance. First, the results provide indication that modeling dependencies is difficult when representing variability. This result is in-line with the findings of Berger et al. [7], according to which the proportions of dependencies in industrial models are relatively low. Modeling tools may support users by providing them with simulation of variability for a specified model (e.g., by representing the valid configurations). This may also help modelers to avoid modeling errors which occurred more often in the hierarchical modeling style. Similar to contemporary theories on human semantic memory [47], future research on variability modeling could also explore higher dimensional (more than 2-D) models in which configurations serve as nodes and similarity connection weights as relations between them. Prior research has already presented a proposal to visualize large feature trees in 3D to avoid scrolling [73]. As soon as models reach a certain size, it also becomes important that tools support users to orientate and navigate through model structures and help them mentally integrate information. Various visualization strategies for displaying hierarchical model structures and interface strategies to navigate between details and their context have been investigated for different types of conceptual models [24,42]. Examples include ‘focus and context’ vs. ‘overview and detail strategy’, or interaction strategies for multiple views (e.g., if items are selected in one view (“brushing”), they are simultaneously selected and highlighted in the other view (“linking”)). Future research could address how such visualization opportunities can be used to support users when interacting with variability models in tools. In addition, adaptation of visualizations to specific user groups might be pursued.

Second, the results reinforce the importance of providing good teaching examples. The choice of examples in tutorials and courses is relevant, as they influence students’ modeling behavior. A fixating, suboptimal example can act as a barrier and be counterproductive to a good model design.

Finally, Table 8 presents the effects of the modeling styles on model comprehension and model creation, based on our experimental results. These effects should be acknowledged when creating variability models either manually or automatically (via tools). Our advice would be to apply constraint-oriented style when creating variability models acknowledging that the hierarchical style has higher risk of errors. However, once the variability model is reasonably established and it is clear that the situation has high choice interdependency there would be comprehension advantages to moving the model into a hierarchical style. Tools should support this transformation, but no such automatic tool exists, yet. Exist-

Table 8
Effects of modeling styles, based on our experimental results.

Task	Dependency	Modeling style	Effect
Model comprehension	Low/high	Hierarchical	• Easier to understand • Lower subjective difficulty • Higher perceived ease of use
	Low	Constrained	• Less errors • Less time
	High	Hierarchical	• Less errors • Less time
Model creation	Low/high	Constrained	• Higher correctness
	Low	Constrained	• A common choice
	High	Hierarchical	• A common choice

ing tools such as Feature IDE[7] provide syntactic support for the variability model notation, and analysis of what configurations are allowed. Commercial products like pure::variants[8] also may provide support for generating the variants (final products) and integrate smoothly with the development environment their customers have, but this is not focused on comprehension as such.

7.3. Threats to validity

There are a number of limitations associated with our experiments that need to be acknowledged. We discuss these limitations next and elaborate on the actions taken to reduce them.

The main sources of weakness to external validity include subjects and materials. Although the participants were students with little experience in modeling, they had the required knowledge and training; thus, we believe that they serve as an adequate proxy for future modelers of variability modeling in general, and CVL in particular. The use of students in experiments similar to ours – not designed for experts – is deemed to be acceptable [41]. Moreover, one should clearly bear in mind that collecting close to a hundred volunteering experts or experienced variability engineers to conduct such an experiment would be prohibitively impractical. Another problem with such a sample would be a possible bias towards one modeling style. Therefore, we deemed it more important to keep the effect of industry experience constant (viz. low). This decision is also reflected by the warning of Gemino and Wand [27, p. 258] that “it is important to recognize that the use of either ‘experienced’ analysts or ‘real’ stakeholders who are very familiar with the application domain, while seemingly providing more realistic conditions, might create substantial difficulties in an experimental study.”

As for the materials used in our experiment, we can encounter threats with respect to models, tasks, the modeling language (CVL), and the tools. Our experiment did use rather small models, and it could be argued that they do not reflect industrial size problems. The tasks needed to be manageable within reasonable time. Even with students, there was a limit as to how complex we could make the task. However, the comprehension tasks contained the complexities that we wanted to investigate. With respect to the modeling task, even though the problem description may seem simple, there were hardly any identical solutions in our sample of created models. We were surprised by the diversity even for semantically correct models, an observation that also supports the need for good style guidelines. Industrial product lines show the same kind of complexities, and although the number of choices will be larger, there are often only more variants per choice, which should not greatly affect the decision on style. Concerning model comprehension, studies have indeed shown that there is an overall negative correlation between higher model size and comprehension [55,62]. While we expect this variable to be an additional independent variable adding to higher intrinsic cognitive load, we do not expect it to interact with the modeling style.

Despite the clear support for the hypothesized associations, the generalizability of findings reported here should be undertaken with caution, as we could only include two different models in the study and we selected a specific variability modeling language – the variability abstraction part of CVL. Moreover, we used a modeling language in which dependencies are expressed in textual constraints and not visually. Visual representation of the dependencies could influence comprehensibility and hence deserve further exploration in the future. As the two models included in the comprehension part and the modeling task were typical representatives, we argue that they provided a reasonable test of comprehensibility, thus assuring construct validity. The selection of the language was done perceiving CVL as an emerging standard that systematically includes the main variability modeling concepts. Regarding standardization, the CVL submission to the OMG was technically recommended, but has not yet been made an OMG technology due to controversies over an American patent and its consequences relating to future commercial tooling for CVL.

With respect to tooling, we applied only one tool in the experiment, and one could imagine that the tool could be biased in favor of one of the modeling styles. The CVL tool used requires that the diagram is built top down, and this could indicate favoring a hierarchical style, but applying a constrained style would only mean that the hierarchy would be shallow. We did not include in our experiments any procedures that would control for this potential mild favoring of the hierarchical style.

To improve conclusion validity, we were assured that random influences to the experimental setting were low. First, participants were committed to the experiment by giving course credit (of about 5%) for participation. Second, the students self-studied CVL, and although conducted in different classes, no influence of the lecturers' capabilities, knowledge, and opinions were introduced to the CVL training.

Although the time taken to complete the whole modeling task was monitored, we could not relate it to the modeling style (as commonly different parts of the model were specified following different modeling styles), nor to the choice interdependency, because participants did work on both basic and extra choices at the same point in time. Thus, we did not include modeling efficiency in the second research model on model creation. Further research might also look at efficiency of creating models in different styles.

8. Conclusions and future research

The present study was primarily designed to determine the effect of modeling style on comprehension and creation of variability models. We further took the choice interdependency into account as an influence factor. Our results are not surprising, as they show that hierarchical (tree) structures are useful in suitable situ-

[7] http://wwwiti.cs.uni-magdeburg.de/iti_db/research/featureide/.
[8] http://www.pure-systems.com/products/pure-variants-9.html.

ations. This obviously was the belief motivating the original FODA approach to define feature trees. Still, our results indicate that expressing constraints through hierarchy is not always the most comprehensible option that modelers currently believe it is. The results showed that the degree of dependency between choices in a model determines what modeling style will be selected when creating a model from natural language descriptions. Furthermore, the degree of dependency between choices also influences the comprehension of the model. Models with high dependency are best understood with hierarchical models, while models with low dependency fit the constrained style. However, modeling in a constrained style leads to fewer modeling errors, independent of the choice interdependency. Thus, while it is more difficult to create hierarchical models, they offer the advantage of higher subjective user acceptance and better comprehension when the model is characterized by high dependency of choices. Summarizing, our study provides further evidence for the utility of cognitive load theory to aid our understanding of cognitive difficulties in variability modeling. These results can be used to generate teaching materials and modeling guidelines.

Another interesting finding was that modelers tended to conform to modeling styles to which they had been previously exposed. However, they did not blindly adhere to these styles, for instance, it occurred more often that they switched from a hierarchical style to a constrained style, rather than vice versa, and their decision of the modeling style was further influenced by the choice interdependency.

Overall, our work denotes an extension to the literature on cognitive aspects of conceptual models for the field of variability modeling, and may ultimately lead to more successful variability modeling and more comprehensible models for managing product lines in practice.

Several opportunities for future research emerge from our study. Particularly, further experimental investigations with a larger variety of models and different types of participants would be required to give a final estimation of the comprehension difficulty of different degrees of choice interdependency. Future studies could also extend this work and examine difficulties in comprehending and modeling variability using other languages, as well as the variability realization part of CVL. Finally, further investigation and experimentation with other modeling styles, their ways of extracting and organizing choices into models, and their implications on comprehension and modeling would be interesting towards a more integrated understanding of cognitive aspects of variability modeling.

Appendix A. The second model used in the experiment.

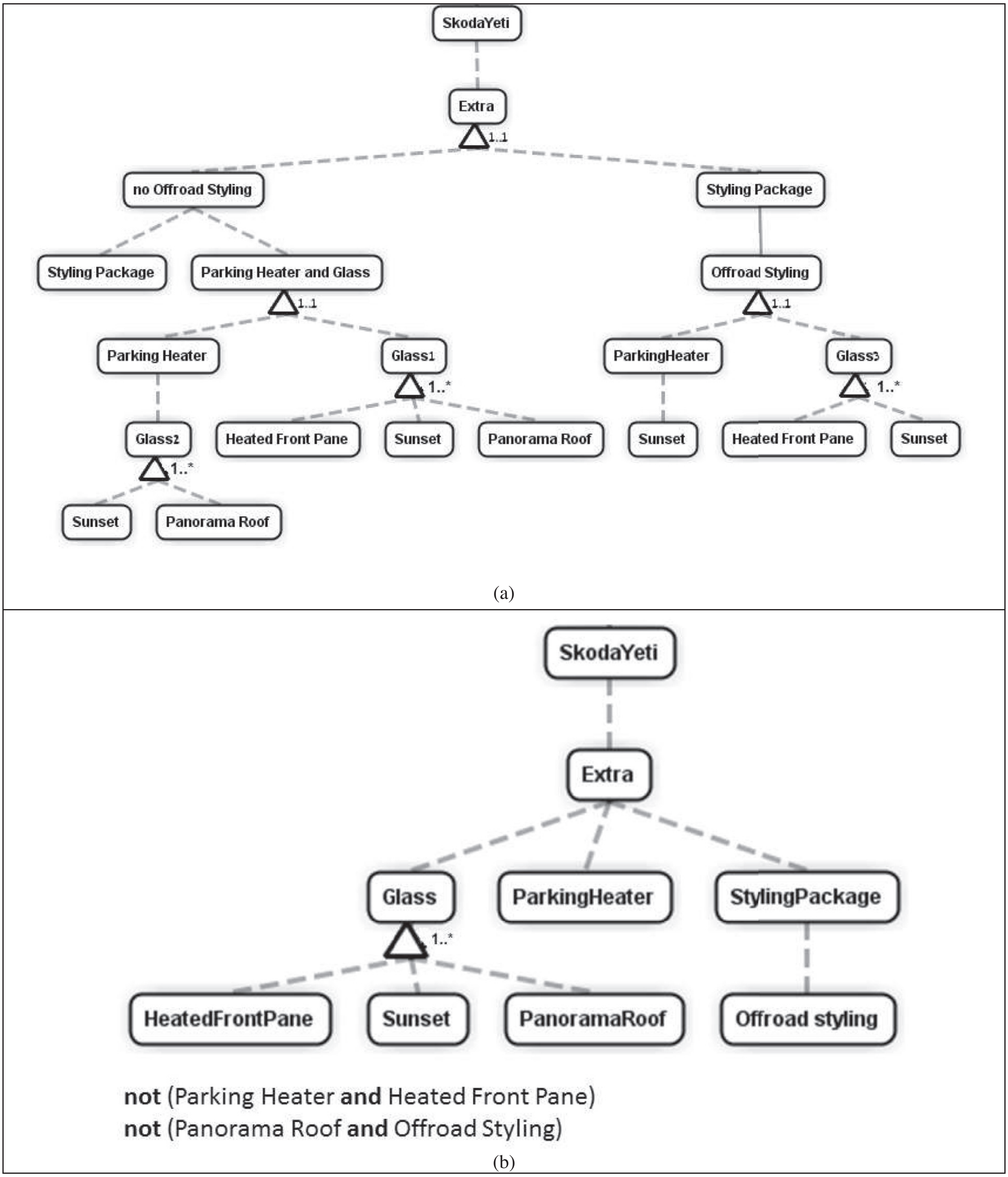

Fig. 5. CVL models specifying the variability within extra choices of Skoda Yeti cars: (a) hierarchical style and (b) constrained style.

Appendix B. Calculation of the dependence index for the first model.

Table 9
Possible configurations for the model depicted in Fig. 1.

Choice ↓	Configuration → 1	2	3	4	5	6
Diesel	T	T			T	T
Benzin			T	T		
Manual	T	T	T		T	
Automatic				T		T
2-wheel-drive	T		T	T		
4×4		T			T	T
Active	T	T	T	T		
Adventure					T	T

T – choice selected, empty – choice deselected.

Table 10
Calculation of the dependence index for the model depicted in Fig. 1.

Choice B ↓	Choice A → Diesel	Benzin	Manual	Automatic	2-wheel-drive	4×4	Active	Adventure
Diesel	X	2	4	4	3	3	2	3
Benzin		X	4	4	3	3	3	3
Manual			X	2	4	4	4	4
Automatic				X	4	4	4	4
2-wheel-drive					X	2	3	3
4×4						X	3	3
Active							X	2
Adventure								X
Sum								91
Max potential sum								112
dependence index								0.81

Appendix C. Comprehension tasks.

"Basic" model:
A Skoda Yeti car can have the following combination of features:

	Correct	Wrong	Cannot be answered from model	I do not know
1. Manual and diesel	○	○	○	○
2. Adventure and benzin	○	○	○	○
3. Automatic and 4×4	○	○	○	○
4. Adventure and 2-wheel-drive	○	○	○	○
5. Active and diesel and automatic	○	○	○	○
6. Diesel and automatic and 4×4	○	○	○	○
7. Active and benzin and 4×4	○	○	○	○
8. Adventure and manual and 4×4	○	○	○	○
9. Active and benzin and manual and 2-wheel-drive	○	○	○	○
10. Automatic and adventure and benzin and 2-wheel-drive	○	○	○	○

"Extra" model:
A Skoda Yeti car can have the following combination of features:

	Correct	Wrong	Cannot be answered from model	I do not know
1. Parking-heater and styling-package	○	○	○	○
2. Panorama-roof and offroad-styling	○	○	○	○
3. Parking-heater and offroad-styling	○	○	○	○
4. Parking-heater and heated-front-pane	○	○	○	○
5. Parking-heater and styling-package and offroad-styling	○	○	○	○
6. Sunset and parking-heater and styling-package	○	○	○	○
7. Heated-front-pane and sunset and panorama-roof	○	○	○	○
8. Sunset and panorama-roof and parking-heater and offroad-styling	○	○	○	○
9. Heated-front-pane and sunset and styling-package and offroad-styling	○	○	○	○
10. Heated-front-pane and sunset and panorama-roof and styling-package	○	○	○	○

Appendix D. The modeling task.

Task description: Skoda Yeti Laurin and Klement

Skoda has a top-of-the-range edition called Laurin and Klement named after the two founders of Skoda, namely, Vaclav Laurin and Vaclav Klement.

Our modeling task focuses on this top-of-the-range edition and on its *diesel cars*.

These cars come with automatic as well as manual gearbox, but when it is automatic, only the 4×4 drive and a 140 hp engine are possible. If the customer opts for a two-wheel drive, s/he must choose the manual shift and a 110 hp engine. The manual shift and the 4×4 drive give the alternatives of both engines (140 hp or 110 hp).

The Laurin and Klement range offers as default a lot of luxury features, but there are still some features that may be selected as extras. The customer can choose parking assistant, backing sensor, double trunk floor or extra wheel. However, choosing the parking assistant excludes choosing the backing sensor.

Appendix E. Supplementary analyses.

E.1. Sub samples

To check whether the type of sub sample used influences our results, we ran some analyses where the sub sample was defined as an additional independent variable. As noted we had four different courses from three universities in our study. As we had used randomization of questionnaires, experimental groups were approximately evenly spread over all sub samples. The results of these analyses are summarized in Table 11 and differences of courses are depicted in Fig. 6.

Adding this new independent variable “sub sample” slightly alters a few results. As would be expected the significance level of the variable familiarity with feature modeling was reduced and now is insignificant. This can be explained by the different amount of education and training on feature modeling at the different universities, which likely leads to differences in self-reported familiarity. The sub-sample was a significant influence factor for comprehension efficiency (time) and subjective difficulty of model. Students of the business modeling course in Vienna took less time for solving the tasks than the other groups and rated the models as more difficult, while the students of the software modeling course in Haifa took most time and rated the models as easiest. It seems possible that the lower time taken is due to “cognitive stopping rules”, which researchers have speculated to lead to minimizing effort in comprehension tasks if tasks are experienced as too difficult to solve [23]. Overall, the results for the courses are in line with the assessment of the researchers that the course in Haifa, whose students received the highest total score on average (87%), prepared students very well in terms of variability modeling, while for the students of the business modeling course in Vienna variability modeling was a completely new field and they performed worst (75%). Due to randomization of experimental conditions, the effects of other influence factors did not change in any relevant way (only slight shifts in decimal places, not a change in significance of effects.)

E.2. Comprehension question type

In a separate analysis, we took a detailed look at the type of comprehension question. To do so, we counted for each comprehension question how often the choices it referred to were mentioned in the model in the hierarchical style (respectively in the model and the textual constraints in the constrained style). In a second step, we subtracted the number of occurrences of choices in the constrained version from the hierarchical model version, resulting in a “redundancy” measure per comprehension question.

Table 11
An overview of the results of the ANCOVAs for repeated measures.

	Effect	F ($df_{\text{Hypothesis}}=84$; $df_{\text{Error}}=1$)	Significance	Partial eta squared
Comprehension effectiveness (total score)	*Sub sample (course)*		n.s.	
	Modeling style	4.08	.05	.05
	Choice interdependency choice interdependency		n.s.	
	Experimental order		n.s.	
	Familiarity with feature modeling	2.85	.10	.04
	*Choice interdependency * experimental order*	4.82	.03	.06
	*Choice interdependency * familiarity with feature modeling*		n.s.	
	*Choice interdependency * modeling style*	37.82	<.001	.32
Comprehension efficiency (time)	*Sub sample (course)*	10.27	.00	.28
	Modeling style		n.s.	
	Choice interdependency	3.62	.06	.07
	Experimental order		n.s.	
	Familiarity with feature modeling		n.s.	
	*Choice interdependency * experimental order*	32.76	<.001	.29
	*Choice interdependency * familiarity with feature modeling*		n.s.	
	*Choice interdependency * modeling style*	43.46	<.001	.35
Perceived ease of use	*Sub sample (course)*		n.s.	
	Modeling style	24.38	<.001	.23
	Choice interdependency		n.s.	
	Experimental order		n.s.	
	Familiarity with feature modeling		n.s.	
	*Choice interdependency * experimental order*		n.s.	
	*Choice interdependency * familiarity with feature modeling*		n.s.	
	*Choice interdependency * modeling style*	6.35	.004	.10
Subjective difficulty of model	*Sub sample (course)*	2.84	.04	.10
	Modeling style	4.36	.04	.05
	Choice interdependency		n.s.	
	Experimental order		n.s.	
	Familiarity with feature modeling		n.s.	
	*Choice interdependency * experimental order*		n.s.	
	*Choice interdependency * familiarity with feature modeling*		n.s.	
	*Choice interdependency * modeling style*		n.s.	

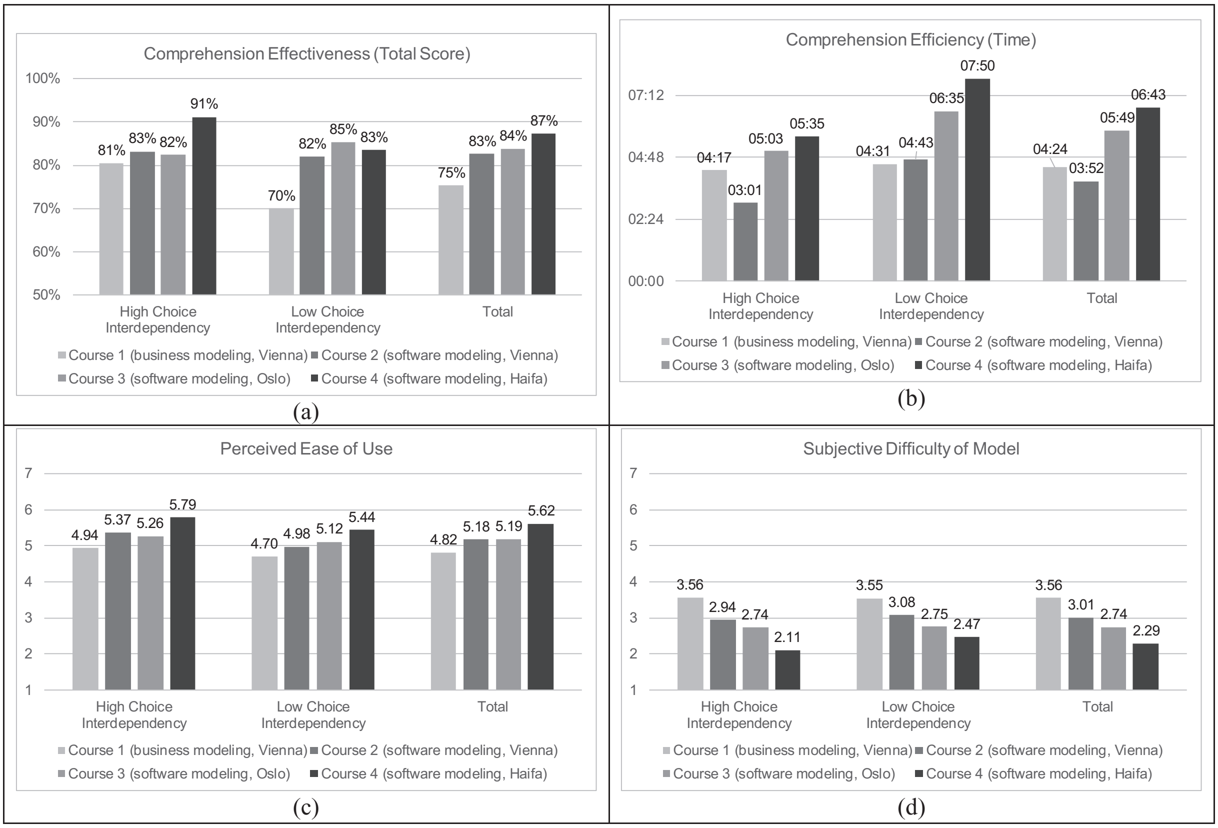

Fig. 6. Results for model comprehension: (a) comprehension effectiveness (total score), (b) comprehension efficiency (time), (c) perceived ease of use, and (d) subjective difficulty of model.

Table 12
An overview of the results of the ANOVAs.

	Hierarchical ($n = 43$)		*Constrained* ($n = 47$)		Total ($N = 90$)		Statistical test
	M (%)	SD	M (%)	SD	M (%)	SD	
Model with high choice dependency – basic choices							
Much higher redundancy in constrained style (4–6)	98	0.09	86	0.28	91	0.22	$F_{df=88} = 6.89$; $p = 0.010$
Higher redundancy in constrained style (2)	92	0.14	66	0.32	79	0.28	$F_{df=88} = 25.22$; $p = 0.000$
Equal redundancy (0)	97	0.10	73	0.30	84	0.26	$F_{df=88} = 24.73$; $p = 0.000$
Model with low choice dependency – extra choices							
Equal redundancy (0)	95	0.21	83	0.38	89	0.32	$F_{df=88} = 3.54$; $p = 0.06$
Lower redundancy in constrained style (−1, −2)	83	0.24	88	0.19	86	0.22	$F_{df=88} = 1.04$; n.s.
Much lower redundancy in constrained style (−3, −4, −5)	67	0.26	80	0.22	74	0.25	$F_{df=88} = 6.56$; $p = 0.01$

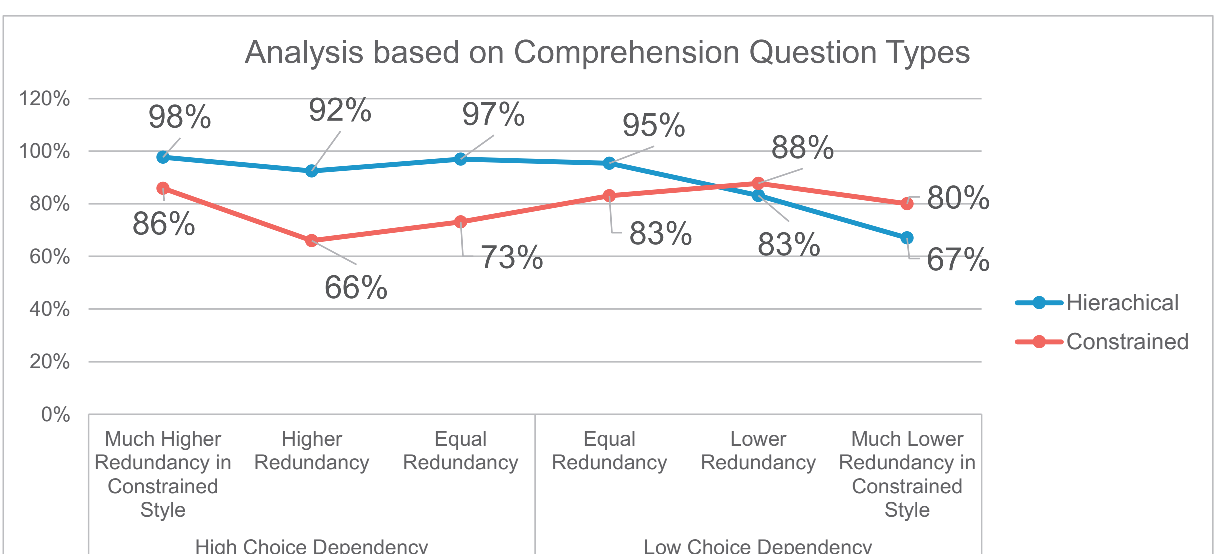

Fig. 7. Redundancy in comprehension question types and comprehension effectiveness.

We grouped questions according to this "redundancy" measure into 3 groups per model (model with basic/extra choices), respectively. The question-based redundancy was in general higher for the model with high choice dependency in the constrained style and for the model with low choice dependency in the hierarchical style. Fig. 5 and Table 11 show an interesting result. It seems that the constrained style outperformed the hierarchical style for comprehension questions that lead to much lower redundancy in the constrained style. In such cases, in which the constrained style leads to equal or higher redundancy, comprehension effectiveness was lower in the constrained style.

References

[1] M. Acher, B. Baudry, P. Heymans, A. Cleve, J.-L. Hainaut, Support for reverse engineering and maintaining feature models, in: Proceedings of the Seventh International Workshop on Variability Modelling of Software-intensive Systems, 2013.
[2] S.W. Ambler, The Elements of UML (TM) 2.0 Style, Cambridge University Press, 2005.
[3] M. Asadi, D. Gasevic, Y. Wand, M. Hatala, Deriving variability patterns in software product lines by ontological considerations, Conceptual Modeling, Springer, 2012, pp. 397–408.
[4] A.D. Baddeley, Working memory, Science 255 (5044) (1992) 556–559.
[5] D. Batory, Feature Models, Grammars, and Propositional Formulas, Springer, 2005.
[6] K. Berg, J. Bishop, D. Muthig, Tracing software product line variability: from problem to solution space, in: Proceedings of the 2005 Annual Research Conference of the South African Institute of Computer Scientists and Information Technologists on it Research in Developing Countries, 2005.
[7] T. Berger, R. Rublack, D. Nair, J.M. Atlee, M. Becker, K. Czarnecki, A. Wąsowski, A survey of variability modeling in industrial practice, in: Proceedings of the Seventh International Workshop on Variability Modelling of Software-intensive Systems, 2013.
[8] F. Bodart, A. Patel, M. Sim, R. Weber, Should optional properties be used in conceptual modelling? A theory and three empirical tests, Inf. Syst. Res. 12 (4) (2001) 384–405.
[9] P.L. Bowen, R.A.O. Farrell, F.H. Rohde, An empirical investigation of end-user query development: the effects of improved model expressiveness vs. complexity, Inf. Syst. Res. 20 (4) (2009) 565–584.
[10] A. Burton-Jones, P. Meso, Conceptualizing systems for understanding: an empirical test of decomposition principles in object-oriented analysis, Inf. Syst. Res. 17 (1) (2006) 38–60.
[11] P. Chandler, J. Sweller, The split-attention effect as a factor in the design of instruction, Br. J. Educ. Psychol. 62 (2) (1992) 233–246.
[12] L. Chen, M. Ali Babar, N. Ali, Variability management in software product lines: a systematic review, in: Proceedings of the Thirteenth International Software Product Line Conference, 2009.
[13] B.T. Christensen, C.D. Schunn, The relationship of analogical distance to analogical function and preinventive structure: the case of engineering design, Mem. Cogn. 35 (1) (2007) 29–38.
[14] R.A. Coll, J.H. Coll, G. Thakur, Graphs and tables: a four-factor experiment, Commun. ACM 37 (4) (1994) 76–86.
[15] K. Czarnecki, U.W. Eisenecker, Feature modeling, Generative Programming – Methods, Tools, and Applications, sixth ed., Addison-Wesley Longman Publishing Co., Inc., 2000.
[16] K. Czarnecki, P. Grünbacher, R. Rabiser, K. Schmid, A. Wąsowski, Cool features and tough decisions: a comparison of variability modeling approaches, in: Proceedings of the Sixth International Workshop on Variability Modeling of Software-Intensive Systems, 2012.
[17] K. Czarnecki, C. Hwan, P. Kim, K. Kalleberg, Feature models are views on ontologies, in: Proceedings of the 2006 International Software Product Line Conference, 2006.
[18] K. Czarnecki, C.H.P. Kim, Cardinality-based feature modeling and constraints: a progress report, in: Proceedings of the 2005 International Workshop on Software Factories at OOPSLA, San Diego, California, USA, 2005.
[19] K. Czarnecki, A. Wasowski, Feature diagrams and logics: There and back again, in: Proceedings of the Eleventh International Software Product Line Conference (SPLC 2007), 2007.
[20] K. Duncker, L.S. Lees, On problem-solving, Psychol. Monogr. 58 (5) (1945) i–113.
[21] M.J. Eppler, A comparison between concept maps, mind maps, conceptual diagrams, and visual metaphors as complementary tools for knowledge construction and sharing, Inf. Vis. 5 (3) (2006) 202–210.
[22] F. Faul, E. Erdfelder, A.-G. Lang, B. Axel, G*Power 3: a flexible statistical power analysis for the social, behavioral, and biomedical sciences, Behav. Res. Methods 39 (2) (2007) 175–191.
[23] K. Figl, Comprehension of procedural visual business process models – a literature review, Bus. Inf. Syst. Eng. 59 (1) (2017) 1–27.
[24] K. Figl, A. Koschmider, S. Kriglstein, Visualising process model hierarchies, in: Proceedings of the 2013 European Conference on Information Systems (ECIS), 2013.
[25] K. Figl, R. Laue, Influence factors for local comprehensibility of process models, Int. J. Hum. Comput. Stud. 82 (2015) 96–110.
[26] M. Galster, D. Weyns, D. Tofan, B. Michalik, P. Avgeriou, Variability in software systems – a systematic literature review, IEEE Trans. Softw. Eng. 40 (3) (2014) 282–306.
[27] A. Gemino, Y. Wand, A framework for empirical evaluation of conceptual modeling techniques, Requir. Eng. 9 (4) (2004) 248–260.
[28] A. Gemino, Y. Wand, Complexity and clarity in conceptual modeling: comparison of mandatory and optional properties, Data Knowl. Eng. 55 (3) (2005) 301–326.
[29] H. Gomaa, Designing software product lines with UML, in: Proceedings of the 2005 SEW Tutorial Notes, 2005.
[30] C. Haisjackl, I. Barba, S. Zugal, P. Soffer, I. Hadar, M. Reichert, J. Pinggera, B. Weber, Understanding declare models: strategies, pitfalls, empirical results, Softw. Syst. Model. 15 (2) (2016) 325–352.
[31] Ø. Haugen, Common variability language (CVL) – OMG revised submission. OMG document ad/2012-08-05, 2012. http://www.omgwiki.org/variability/lib/exe/fetch.php?media=cvl-revised-submission.pdf.
[32] Ø. Haugen, O. Øgård, BVR – better variability results, System Analysis and Modeling: Models and Reusability, Springer, 2014, pp. 1–15.
[33] C. Houy, P. Fettke, P. Loos, Understanding understandability of conceptual models – What are we actually talking about? in: P. Atzeni, D. Cheung, S. Ram (Eds.), Conceptual Modeling, 7532, Springer, Berlin Heidelberg, 2012, pp. 64–77.
[34] N. Itzik, I. Reinhartz-Berger, Generating feature models from requirements: structural vs. functional perspectives, in: Proceedings of the Eighteenth International Software Product Line Conference: Companion Volume for Workshops, Demonstrations and Tools-Volume 2, 2014.
[35] N. Itzik, I. Reinhartz-Berger, Y. Wand, Variability analysis of requirements: considering behavioral differences and reflecting stakeholders perspectives, Trans. Softw. Eng. 42 (7) (2016) 687–706.
[36] D.G. Jansson, S.M. Smith, Design fixation, Des. Stud. 12 (1) (1991) 3–11.
[37] S. Kalyuga, P. Ayres, P. Chandler, J. Sweller, The expertise reversal effect, Educ. Psychol. 38 (2003) 23–31.
[38] K.C. Kang, S.G. Cohen, J.A. Hess, W.E. Novak, A.S. Peterson, Feature-Oriented Domain Analysis (FODA) Feasibility Study, Software Engineering Institute, Carnegie Mellon University, 1990.
[39] K.C. Kang, H. Lee, Variability modeling, Systems and Software Variability Management, Springer, 2013, pp. 25–42.
[40] P.A. Kirschner, Cognitive load theory: implications of cognitive load theory on the design of learning, Learn. Instr. 12 (1) (2002) 1–10.
[41] B. Kitchenham, S.L. Pfleeger, L.M. Pickard, P.W. Jones, D.C. Hoaglin, K. El Emam, J. Rosenberg, Preliminary guidelines for empirical research in software engineering, IEEE Trans. Softw. Eng. 28 (8) (2002) 721–734.
[42] A. Koschmider, S. Kriglstein, M. Ullrich, Investigations on user preferences of the alignment of process activities, objects and roles, in: Proceedings of the 2013 Conferences On the Move to Meaningful Internet Systems (OTM 2013), 2013.
[43] J.H. Larkin, H.A. Simon, Why a diagram is (sometimes) worth ten thousand words, Cogn. Sci. 11 (1) (1987) 65–100.
[44] K. Lee, K.C. Kang and J. Lee, Concepts and guidelines of feature modeling for product line software engineering, Software Reuse: Methods, Techniques, and Tools, LNCS 2319, 2002, pp. 62–77.
[45] A. Maes, G. Poels, Evaluating quality of conceptual modelling scripts based on user perceptions, Data Knowl. Eng. 63 (3) (2007) 701–724.
[46] R.E. Mayer, R. Moreno, Nine ways to reduce cognitive load in multimedia learning, Educ. Psychol. 38 (1) (2003) 43–52.
[47] T.P. McNamara, J.B. Holbrook, Semantic memory and priming, Handbook of Psychology, John Wiley & Sons, Inc, 2003.
[48] D.L. Moody, The "physics" of notations: towards a scientific basis for constructing visual notations in software engineering, IEEE Trans. Softw. Eng. 35 (5) (2009) 756–779.
[49] J. Parsons, An experimental study of the effects of representing property precedence on the comprehension of conceptual schemas, J. Assoc. Inf. Syst. 12 (6) (2011) 401–422.
[50] J. Parsons, L. Cole, What do the pictures mean? Guidelines for experimental evaluation of representation fidelity in diagrammatical conceptual modeling techniques, Data Knowl. Eng. 55 (3) (2005) 327–342.
[51] J. Parsons and Y. Wand, Classification in systems design: exploiting the survival value of categories, 2006 https://pdfs.semanticscholar.org/2abe/44fedd1ef905b3e07e562bbf9869fe3cbc3c.pdf.
[52] J. Parsons, Y. Wand, Extending classification principles from information modeling to other disciplines, J. Assoc. Inf. Syst. 14 (3) (2013).
[53] K. Pohl, G. Böckle, F. van der Linden, Software product line engineering: foundations, Principles and Techniques, Springer, 2005.
[54] J. Recker, Continued use of process modeling grammars: the impact of individual difference factors, Eur. J. Inf. Syst. 19 (1) (2010) 76–92.
[55] J. Recker, Empirical investigation of the usefulness of Gateway constructs in process models, Eur. J. Inf. Syst. 22 (6) (2013) 673–689.
[56] I. Reinhartz-Berger, K. Figl, Comprehensibility of orthogonal variability modeling languages: the cases of CVL and OVM, in: Proceedings of the Eighteenth Software Product Line Conference, Florence, Italy, 2014.
[57] I. Reinhartz-Berger, K. Figl, Ø. Haugen, Comprehensibility of feature models: experimenting with CVL, in: Proceedings of the ACM/IEEE Seventeenth International Conference on Model Driven Engineering Languages and Systems (MODELS), Valencia, Spain, 2014.

[58] I. Reinhartz-Berger, A. Sturm, Enhancing UML models: a domain analysis approach, Sel. Read. Database Technol. Appl. (2008) 330.
[59] I. Reinhartz-Berger, A. Tsoury, Experimenting with the Comprehension of feature-oriented and UML-based core assets, in: T. Halpin, S. Nurcan, J. Krogstie, P. Soffer, E. Proper, R. Schmidt, I. Bider (Eds.), Enterprise, Business-Process and Information Systems Modeling, 81, Springer, Berlin Heidelberg, 2011, pp. 468–482.
[60] R.S. Rist, Schema creation in programming, Cogn. Sci. 13 (3) (1989) 389–414.
[61] L. Rokach, O. Maimon, Top-down induction of decision trees classifiers – a survey, IEEE Trans. Syst. Man Cybern. Part C (Appl. Rev.) 35 (4) (2005) 476–487.
[62] L. Sánchez-González, F. García, J. Mendling, F. Ruiz, Quality assessment of business process models based on thresholds, in: R. Meersman, T. Dillon, P. Herrero (Eds.), On the Move to Meaningful Internet Systems: OTM 2010, 6426, Springer, Berlin Heidelberg, 2010, pp. 78–95.
[63] E. Schmider, M. Ziegler, E. Danay, L. Beyer, M. Bühner, Is it really robust, Methodology 6 (2010) 147–151.
[64] G. Shanks, E. Tansley, J. Nuredini, D. Tobin, R. Weber, Representing part–whole relations in conceptual modeling: an empirical evaluation, MIS Q. 32 (3) (2008) 553–573.
[65] S. She, R. Lotufo, T. Berger, A. Wasowski, K. Czarnecki, Reverse engineering feature models, in: Proceedings of the Thirty-third International Conference on Software Engineering (ICSE), 2011.
[66] S.M. Smith, T. Ward, J. Schumacher, Constraining effects of examples in a creative generation task, Mem. Cogn. 21 (6) (1993) 837–845.
[67] P. Soffer, M. Kaner, Y. Wand, Towards understanding the process of process modeling: theoretical and empirical considerations, in: Proceedings of the 2012 Business Process Management Workshops, 2012.
[68] R.J. Sternberg, Cognitive psychology, Wadsworth, Cengage Learning, Belmont, 2009.
[69] J. Sweller, Cognitive load during problem solving: effects on learning, Cogn. Sci. A Multidiscip. J. 12 (2) (1988) 257–285.
[70] J. Sweller, Element interactivity and intrinsic, extraneous, and germane cognitive load, Educ. Psychol. Rev. 22 (2) (2010) 123–138.
[71] B.G. Tabachnick, L.S. Fidell, Using Multivariate Statistics, Pearson Education, Inc., Boston, 2007.
[72] T. Thum, C. Kastner, S. Erdweg, N. Siegmund, Abstract features in feature modeling, in: Proceedings of the Fifteenth International Software Product Line Conference (SPLC), 2011.
[73] P. Trinidad, A.R. Cortés, D. Benavides, S. Segura, Three-dimensional feature diagrams visualization, in: Proceedings of the 2008 Software Product Line Conference (SPLC), 2008.
[74] O. Turetken, T. Rompen, I. Vanderfeesten, A. Dikici, J. van Moll, The effect of modularity representation and presentation medium on the understandability of business process models in BPMN, in: Proceedings of the 2016 International Conference on Business Process Management, 2016.
[75] Y. Wand, R. Weber, An ontological model of an information system, IEEE Trans. Softw. Eng. 16 (11) (1990) 1282–1292.
[76] Y. Wand, R. Weber, On the ontological expressiveness of information systems analysis and design grammars, J. Inf. Syst. 3 (4) (1993) 217–237.
[77] Y. Wand, R. Weber, On the deep structure of information systems, Inf. Syst. J. 5 (3) (1995) 203–223.
[78] S. Zugal, P. Soffer, C. Haisjackl, J. Pinggera, M. Reichert, B. Weber, Investigating expressiveness and understandability of hierarchy in declarative business process models, Softw. Syst. Model. 14 (3) (2015) 1081–1103.