



Evaluating user acceptance of knowledge-intensive business process modeling languages

Amin Jalali¹

Received: 27 February 2022 / Revised: 18 April 2023 / Accepted: 17 July 2023 / Published online: 2 August 2023
© The Author(s) 2023

Abstract

Case Management has been evolving to support knowledge-intensive business process management, resulting in different modeling languages, e.g., Declare, Dynamic Condition Response (DCR), and Case Management Model and Notation (CMMN). A language will die if users do not accept and use it in practice—similar to extinct human languages. Thus, evaluating how users perceive languages is important to improve them. Although some studies have investigated how the process designers perceived Declare and DCR, there is a lack of research on how they perceive CMMN—especially in comparison with other languages. Therefore, this paper investigates and compares how process designers perceive these languages based on the Technology Acceptance Model. The paper includes two studies conducted in 2020 and 2022, both performed by educating participants through a course, with feedback on their assignments, to reduce biases. The perceptions are collected through questionnaires before and after feedback on the final practice. Results show that the perceptions change is insignificant after feedback due to the participants being well-trained. The reliability of responses was tested using Cronbach’s alpha. The results of the first study show that both DCR and CMMN were perceived as having acceptable usefulness and ease of use, but CMMN was perceived as significantly better than DCR in terms of ease of use. The results of the second study show that only DCR was perceived significantly better than Declare in terms of usefulness. The participants’ feedback shows potential areas for improvement in languages and tool support to enhance perceived usefulness and ease of use.

Keywords Business process modeling · Knowledge-intensive · Case management · CMMN · DCR · Declare

1 Introduction

Case Management is a research paradigm that supports the management of knowledge-intensive processes (KiP) [16]. The process is defined around the case concept, such as a patient in healthcare or a customer in the insurance domain. Knowledge workers, rather than pre-defined rigid rules, decide how a case should proceed in these processes. As a result, the support for KiPs differs from workflow-based processes. This difference can also be observed in the management lifecycle, known as the collaborative knowledge work lifecycle [16], which deviates from the typical workflow-based management lifecycle.

The collaborative nature of work in KiPs requires more freedom for knowledge workers to decide how a case shall proceed. Therefore, traditional workflow-based models are not suitable for case management because: (i) the support shall be limited to specific variants which might be counter-productive for other variants [9], or (ii) the process model becomes too complex by capturing all variants required by different cases, a.k.a. spaghetti models. To address this, several case management modeling languages have been defined, e.g., Declare [40, 42], Dynamic Condition Response (DCR) [23], and Case Management Model and Notation (CMMN) [38], which is based on Guard-Stage-Milestone (GSM) [24]. However, as these languages are new, it is uncertain whether users will accept them, leaving room for further investigation.

The user acceptance of process modeling languages, like other information systems, can be evaluated based on two variables: perceived usefulness (PU) and perceived ease of use (PEU) [14]. Evaluating these variables is very important as they enable us to predict the usability of modeling

Communicated by Editorial Responsibility: Selmin Nurcan, Rainer Schmidt and Adriano Augusto.

✉ Amin Jalali
aj@dsv.su.se

¹ Stockholm University, Forum 100, 164 40 Kista, Sweden

languages and compare them to each other. There are few studies on how users perceive Declare and DCR, e.g. [20, 44, 54], yet there is a gap in research in comparing different notations in terms of understandability [49].

Therefore, this paper aims to evaluate the user acceptance of CMMN, DCR, and Declare languages and compare them. The user acceptance is evaluated by applying the Technology Acceptance Model [14]. As these languages are new, finding users who already know them is not applicable. Thus, the users shall be trained, and the acceptance shall be measured based on their perception (self-assessment). As the self-assessment is a subjective score, it may differ during the learning process due to biased factors such as over- or under-confidence [17]. These biased factors can be minimized by repeated experiences and feedback [17]. Thus, it is important to check whether user perceptions are stable and whether the biased factors are minimized.

The *overall* users' perceptions before and after receiving the feedback for DCR and CMMN, as two industrial-based languages, were measured through a study in 2020 and presented in [28]. This paper extends the result by analyzing data and comparing how these two languages are perceived in comparison with each other, answering these questions:

- R.Q.1. How do trained process designers perceive the usefulness and ease of use of CMMN and DCR compared to each other as industrial-based languages?
- R.Q.2. Has the feedback significantly changed how trained process designers perceived the usefulness and ease of use of each language?

In addition, this paper extends the previous work by repeating the study in 2022, including the Declare language, and involving only participants with industrial working experience, answering this question:

- R.Q.3. How do trained process designers with industrial working experience perceive the usefulness and ease of use of CMMN, DCR, and Declare compared to each other?

The previous paper did not investigate: (i) how participants perceived each language separately, (ii) if there were any significant differences between how participants perceived usefulness and ease of use of CMMN compared with DCR, and (iii) if the feedback significantly changed how participants perceived each language. This paper addresses these gaps by extending the previous publication and presents a more in-depth literature review of papers that performed user evaluation for CMMN or DCR. In addition, it performs a new comparison between CMMN, DCR, and Declare by repeating the study in 2022 as mentioned earlier.

Participants were trained in the business process and case management course at Stockholm University. This course is part of a distance program taken by students from different countries, and it is common for students to work in the industry in parallel. The students practiced these languages and received feedback on their assignments in order to minimize the biases of overconfidence and underconfidence. Then, we collected perceptions before and after the feedback on their performance on the examination. Participation in this study was optional.

The significant differences are tested through nonparametric statistical significance tests, and the reliability of responses is tested using Cronbach's alpha. The results of the first study show that both DCR and CMMN were perceived as having acceptable usefulness and ease of use, but CMMN was perceived as significantly better than DCR in terms of ease of use. The results of the second study show that DCR was perceived significantly better than Declare in terms of usefulness. The comparison of results on users' perceptions indicates potential reasons for differences, including the importance of supporting interactive simulation by the tool when learning a language. The tool's simulation enabled participants in the second study to point out an issue they found hindering the usability.

The remainder of this paper is structured as follows: Section 2 gives a brief background on related work. Section 3 describes the method that is used in this study. Section 4 reports the result and discussion, and Sect. 5 concludes the paper.

2 Background

This section briefly summarizes related work that performed user evaluation for KiPs. It also gives an overview of related work that performed user evaluation for CMMN and DCR through a limited scoping literature review. In addition, it gives excerpts of DCR, CMMN, and Declare notations. Please note that we do not aim to give the full syntax of these languages, which can be found in related literature.

2.1 Users perceptions in business process management

A language will eventually die if people do not accept and use it in practice, which is also true for business process modeling languages. Thus, it is important to evaluate how users perceive languages to determine whether there is a need for improvement. The evaluation can help us to improve process modeling languages. Here, we mention some related works that evaluated user acceptance in the Business Process Management (BPM) area, in general, and in the case management area, in particular.

2.1.1 Users perceptions in business process modeling

Process models can quickly become complex as they represent complex business processes in practice. Therefore, different approaches have been developed to enable process designers to deal with the complexity. La Rosa et al. categorize these approaches into two main categories, i.e., concrete syntax modifications [31] and abstract syntax modifications [32]. They also identified different patterns that can be applied in each category.

The concrete syntax modifications refer to (i) using highlights, (ii) following layout guidelines, (iii) following naming guidelines, or (iv) applying different representations techniques, e.g., using icons for tasks, etc. [31]. The abstract syntax modifications refer to applying different abstraction techniques in business process modeling, i.e., vertical, horizontal, or orthogonal modularization techniques [32]. La Rosa et al. evaluated how users perceived the identified patterns' usefulness and ease of use by applying the technology acceptance model [14]. Their evaluation study showed that all identified patterns are perceived as relevant.

Evaluating the users' perception is important because the artifact's actual usage is influenced by the potential users' perceptions—which can be measured in terms of usefulness and ease of use [14]. Therefore, researchers have used techniques like the technology acceptance model to evaluate different business process modeling techniques. For example, the technology acceptance model is used to evaluate (i) how users perceived orthogonal modularization based on aspect-oriented business process modeling in [27, 29], (ii) how users perceived the vertical decomposition using BPMN in [50], and (iii) how graphical highlights can increase the cognitive effectiveness of business process models in [30].

2.1.2 Users perceptions in case management

Case management is a relatively new paradigm compared to Business Process Management, and few languages have been developed to support managing cases. Declarative Service Flow Language (DecSerFlow) [51] (a.k.a., Declare) can be considered as one of the first attempts to develop such languages.

The understandability and maintainability of process models are crucial for any process modeling language. Thus, Fahland et al. identified and hypothesized a set of propositions describing differences between imperative and declarative process modeling languages based on understandability and maintainability in [18] and [19], respectively. Weber et al. [52] conducted a controlled experiment to investigate whether process designers can deal with increased levels of constraints when using Declare. The participants were 41 students from two universities. The results show that the par-

ticipants can deal with introduced constraints, which justifies further declarative process modeling development.

Pichler et al. investigated whether the imperative or the declarative process modeling languages are better understood by running an experiment with 27 students [43]. Their study shows that students understand imperative languages better.

Zugal et al. [54] investigated how hierarchies affect the expressiveness and understandability of declarative models. The study is based on nine participants from two universities. It shows that hierarchies enhance expressiveness. It also shows that the hierarchies can increase the models' understandability, but they should be applied with care. Haisjackl et al. [20] investigated how declarative models are understood through an explorative study. Their study shows that the subjects could understand a single constraint well, but it was challenging for them to handle a combination of constraints. This study also shows that some graphical notations in Declare, similar to imperative modeling languages, cause considerable trouble in understandability.

2.2 Users perceptions for CMMN and DCR

This paper primarily focuses on industrial-based case management modeling languages. Thus, a limited scoping literature review is conducted in this study to identify related work that reported user evaluation results for CMMN and DCR, which is another industrial-based language. The result shows the lack of such evaluation. Thus, it can be concluded that the evaluation of the combination of CMMN, DCR, and Declare is also missing—as a subset of the above combination. Therefore, this section explains the literature review process and presents the result.

The literature review process includes three steps, i.e., finding the papers mentioned CMMN or DCR, filtering the relevant papers, and filtering the papers with evaluation results. The result of each step is demonstrated in Fig. 1.

2.2.1 Literature review process

The *first step* starts by searching Web of Science (Clarivate) and Scopus using two keywords, i.e., "case management model and notation" and "Dynamic Condition Response." The search was performed on all fields on 2022-02-11. The result was 400 papers, where the total number of papers per each database per each keyword is presented on the left side of Fig. 1. Twenty and 230 papers were found for CMMN in Web of Science and Scopus, respectively. Also, 18 and 200 papers were found for DCR in Web of Science and Scopus, respectively. As shown in Fig. 1, 33 papers mentioned both CMMN and DCR—all indexed in Scopus.

The *second step* was about filtering papers that are about CMMN or DCR. In this step, we excluded papers which

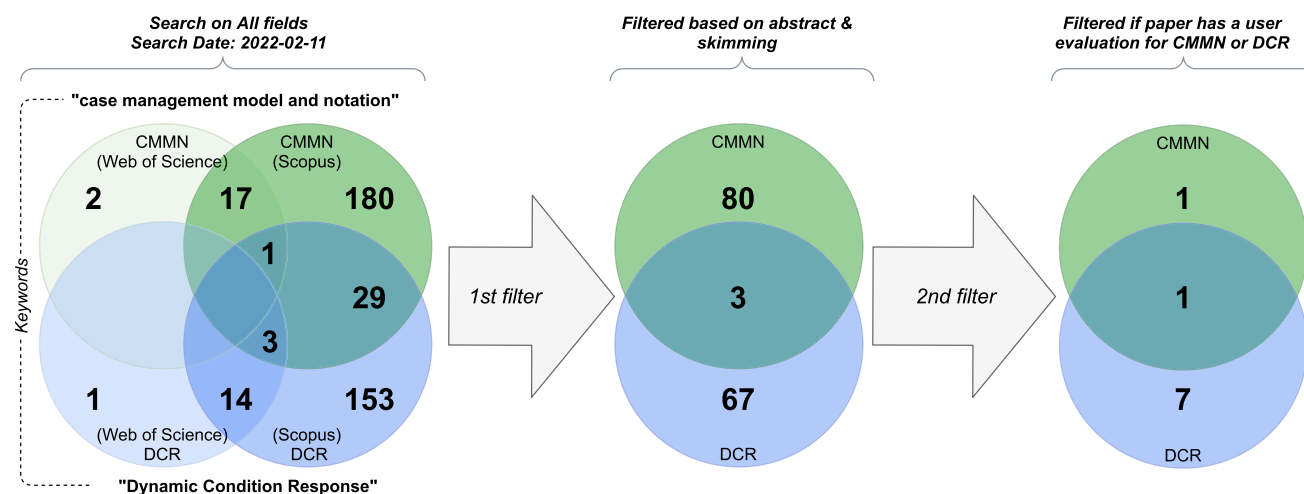


Fig. 1 The scoping review result for each step

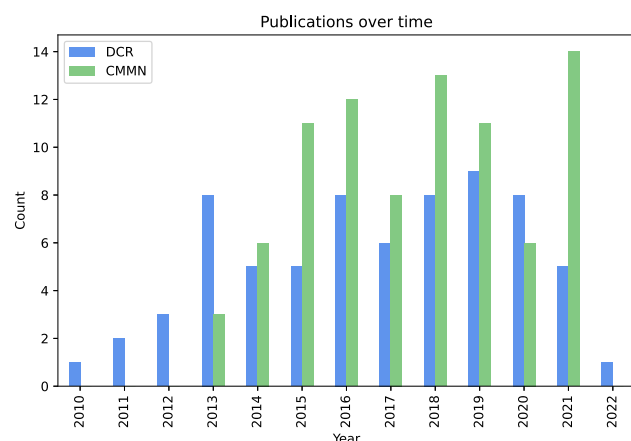


Fig. 2 The trend of related publications over time

were only mentioned these languages in the related work section. Also, we exclude literature review papers that only summarize other papers. The papers are filtered based on their title, abstracts, and keywords. Also, 329 papers are skimmed to make the filter more precise. The result was 150 papers about CMMN or DCR, among which only 3 papers were about both languages. The result of this step is shown in the middle of Fig. 1.

Figure 2 shows the distribution of the papers which are filtered in the second step. It shall be noted that the number of publications for 2022 is not complete, as we performed the search at the beginning of the year.

The *third step* was about filtering papers that reported the user evaluation result for CMMN or DCR. The result was 9 papers, among which only 1 paper was about both CMMN and DCR, shown on the right side of Fig. 1. It shall be noted that the paper which evaluated both CMMN and DCR is the conference paper based on which this work is extended [28].

None of these papers reported any user evaluation comparing the two notations.

2.2.2 Literature review result

Table 1 lists papers that evaluated CMMN or DCR with the help of users. As can be seen, few studies involve users in evaluating these notations. Among these studies, there are very few studies that evaluate user perceptions. The papers are listed chronologically in the table.

Reijers, H.A. et al. [44] evaluated DCR and Declare through a workshop with 10 industrial participants. They divided practitioners into two groups. One group worked with DCR, and the other group worked with Declare. The evaluation is performed using both quantitative and qualitative approaches. As a result of quantitative evaluation, the paper reports the perceived usefulness and the perceived ease of use for these two languages. The reliability of items is measured by Cronbach's alpha. This study shows no significant difference in how participants perceived DCR and Declare. The result also shows that the average perception of participants was positive. This study also revealed some insights about declarative and hybrid approaches resulting from qualitative evaluation.

Marquard, M. et al. [36] evaluated the understandability of concepts and elements of DCR language through a tutorial which is organized at the BPM conference. The evaluation is performed by 1 industrial and 11 academic participants. Five and nine participants had experience with DCR and Declare before the tutorial, respectively. The study shows how much different elements of DCR are understandable by participants. It also measures four elements capturing the usability of the tool, i.e., Modeling Screen, Adding Friends, Individual simulation, and Collaborative simulation.

Table 1 The list of papers that performed user evaluation for CMMN or DCR

| Paper | Year | Language | | Evaluation Approach | | Number of Participants | | Languages comparison | user perception |
|-------|------|----------|-------------|---------------------|--------------|------------------------|----------|----------------------|-----------------|
| | | CMMN | DCR | Qualitative | Quantitative | Industrial | Academic | | |
| [44] | 2013 | - | + & Declare | + | + | 10 | 0 | + | + |
| [36] | 2015 | - | + | - | + | 1 | 11 | - | + |
| [1] | 2019 | - | + | + | - | 7 | 10 | - | - |
| [5] | 2019 | - | + | + | - | 5 | 10 | - | - |
| [15] | 2020 | - | + | + | - | 1 | 0 | - | - |
| [6] | 2020 | - | + | + | - | 2 | 11 | - | - |
| [2]* | 2021 | - | + | + | - | 5 | 10 | - | - |
| [45] | 2021 | + | - | - | + | 24 | 24 | - | + |
| [28] | 2021 | + | + | - | + | 0 | 20 | - | + |

* Paper [2] extends paper [5]

Andaloussi, A.A. et al. [1] investigated the effect of using text highlighter with DCR through an exploratory study that includes 7 and 10 industrial and academic participants, respectively. DCR text highlighter is an additional tool that enables process modelers to identify different elements of a business process and relate them to the model. The study used a qualitative approach, and it indirectly interprets user perceptions. Indeed, the users' data are collected as they use the toolset. This work does not evaluate user perception, but it is included in the list of papers as it evaluates the tools by involving users and analyzing their data. They conclude that the "highlighter was perceived more efficient to identify and append activities and roles to the model" [1].

Andaloussi, A.A. et al. [2, 5] conducted an exploratory study investigating the benefits and challenges in modeling business processes with the help of three artifacts, i.e., a process model, textual annotations, and an interactive simulation. The study includes 5 and 10 industrial and academic participants, respectively. Four industrial participants did not have any background in process modeling, so they participated as domain experts. Academic participants were familiar with process modeling. The study collected data through eye-tracking software, based on which they analyzed how users performed the modeling. The result shows how these groups have utilized the artifacts in different ways. Also, it shows how the artifacts are used differently for different tasks. This paper represents the potential of using eye-tracking techniques to understand users' needs in process modeling.

Debois, S. et al. [15] explained how DCR supports compliant-by-design modeling. Although this paper does not focus specifically on the evaluation, it evaluated the proposed approach by interviewing an expert from the industry. The result shows the approach's relevancy, usability, and limitations. The subject described the approach as helpful, and the study demonstrates how DCR can help better compliance with the law in modeling business processes.

Andaloussi, A.A. et al. [6] identified a set of quality dimensions for DCR Graph through a qualitative study involving 2 industrial and 11 academic participants. The study follows the Personal Construct Psychology (PCT) theory. The finding can improve our understanding of how experts can evaluate quality in the DCR model in particular and declarative models in general.

Routis, I. et al., [45] evaluated the usefulness, ease of use, attitude of using, and intention to adopt CMMN through a workshop. These variables are measured for each element of CMMN based on which the overall perception is calculated. The workshop took 4 weeks, and it involved 24 participants. The participants are described as process modelers or process engineers in the paper. These terms can describe the roles of participants in the study. The demography of participants is not documented. Thus, it is not clear if they are industrial or academic participants, so they cannot be explicitly distinguished according to the Table 1's format.

Jalali, A. [28] evaluated how users perceive the usefulness and ease of use of DCR and CMMN. The perception is measured by applying Technology Acceptance Model—similar to [44]. The study includes 20 master student participants trained in a course, and the result is reported quantitatively.

Some papers do not perform user evaluation, so they are excluded from Table 1. However, some of them report interesting results, which are worth mentioning here. Ioannis, R. et al. evaluated the applicability of CMMN through a case study [46, 47]. This study includes 6 industrial participants divided into two groups and modeled real-case scenarios using CMMN. The result revealed two modeling styles used by each group.

2.3 Dynamic condition response (DCR)

Hildebrandt and Mukkamala introduced DCR in 2010 as a declarative process modeling language [21]. The syntax of the language includes the definition of nodes (the group of

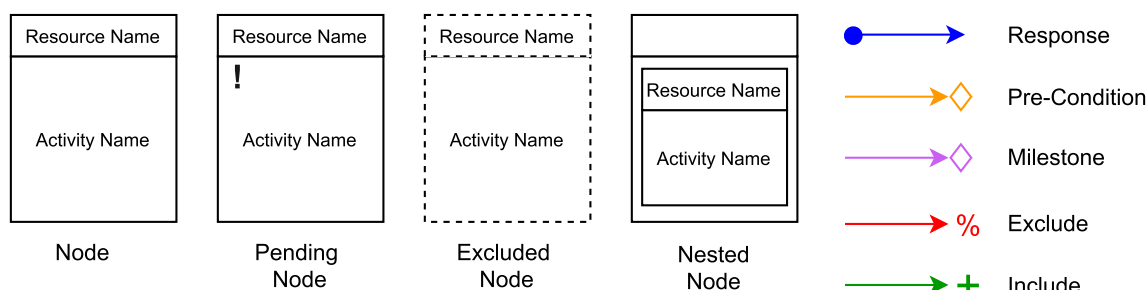


Fig. 3 An excerpt of DCR Syntax

an activity and its roles) and the relation that can be defined among them, i.e., response ($\bullet \rightarrow$), condition ($\rightarrow \bullet$), inclusion ($\rightarrow +$), and exclusion ($\rightarrow \%$) [21].

They also defined semantics for DCR, where several events can occur for a node. A node can be included or excluded from the process structure. A node can also be in the pending state, meaning that the process cannot successfully be finished until an event of the node occurs. In short, the response relation among nodes a and b ($a \bullet \rightarrow b$) means that the state of node b will be pending if an event of node a happens. More precisely, event b must eventually happen if event a happens. The condition relation among nodes a and b ($a \rightarrow \bullet b$) means that an event of b cannot occur unless a occurs.

The DCR's syntax and semantics have evolved over the years. For example, the syntax is enriched to represent the excluded nodes by dashed border line [23]. The extended syntax also represents a node's pending state by decorating it with an exclamation mark (!). In addition, milestone relation ($\rightarrow \diamond$) and nested nodes are also introduced to the language [22]. The milestone relation among two nodes a and b ($a \rightarrow \diamond b$) means that b can occur as long as a is not in the pending state.

DCR graph is the toolset that enables the modeling and simulation of DCR models. Currently, it supports other relations, including pre-condition. The pre-condition relation among two nodes a and b means that $a \rightarrow \diamond b \wedge a \rightarrow \bullet b$. This relation is represented with the same graphical notation as the milestone but with a different color. Figure 3 shows an excerpt of DCR syntax.

2.4 Case management model and notation (CMMN)

Case Management Model and Notation (CMMN) is a case modeling language that is defined by Object Management Group (OMG) [38]. This language is developed by extending the Guard-Stage-Milestone (GSM) language [24, 25]. Figure 4 shows an excerpt of CMMN syntax.

The *case plan model*, represented by a folder, is the core part of a CMMN model. The *case plan model* captures the

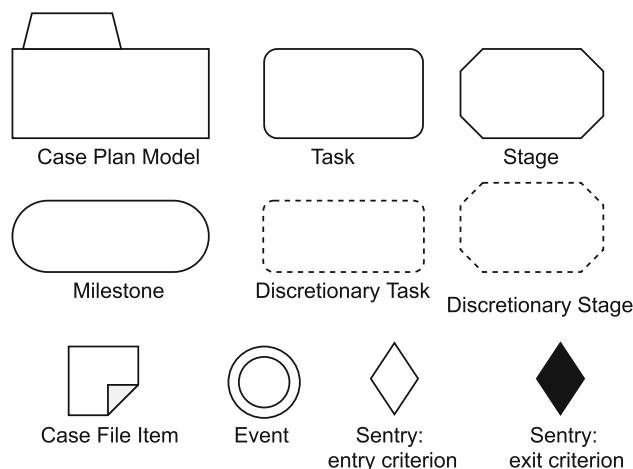


Fig. 4 CMMN Syntax—basic elements

complete behavior of a case, and all other elements will be children of the *case plan model*. The *case file item* represents the data, *Task* represents activities that can happen, *Stage* represents a container that includes other elements (like sub-process in BPMN), *Milestone* represents an achievable target, and *event* represents something that happens during the course of a case. Some elements, like tasks, can have sentries on their border. A *sentry* can represent *entry criterion* or *exit criterion*, which defines the condition or event—based on which the element can be enabled or terminated, respectively.

Tasks and Stages can also be represented by the dashed borderline, which is known as *discretionary task* or *discretionary stage*. Discretionary elements are not available to knowledge workers at runtime. However, they can add these elements to their case plan at runtime. These elements can be related to each other through lines, and the connection rules are defined in CMMN specification [38]. Elements can also be decorated using different decoration icons. For example, ! or # indicates that the task, stage, or milestone are mandatory or repetitive, respectively. As another example, ▷ indicates that a task or stage shall be activated manually at runtime.

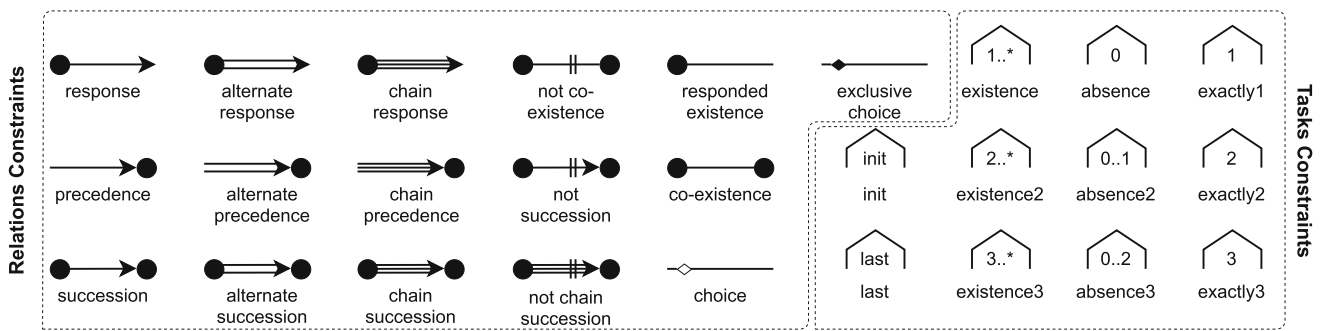


Fig. 5 Declare Syntax—based on CPN Tools

2.5 Declare

Declare was named for a prototype of a Constraint-Based Workflow Management System based on ConDec process modeling language - proposed by Maja Pešić [40]. This process modeling language is called Declare later by the community, e.g., [10, 34, 35].

Declare raised the idea that it is possible to specify what is forbidden in process models via a rule-based syntax, so the system supports both allowed and optional scenarios [40]. This idea suits describing more flexible processes—as describing all possible paths will result in a spaghetti model.

Declare Designer is the first editor supporting modeling process models using Declare, and Declare Service is developed as an extension of YAWL to execute such models [41]. CPN Tools is also extended to support modeling and simulating Declare models in an interactive way [53], which also enables the composition of hybrid models combining Declare and Coloured Petri nets.

ProM and RuM are two software enabling the discovery of Declare process models from log files using different algorithms, e.g., [3, 33]. They also enable checking the conformance of a Declare model and given event logs, e.g., [3, 11].

Tasks are represented using rectangles in Declare, and all tasks can be executed in any order unless there is a constraint prohibiting the execution. Figure 5 shows the list of constraints according to the CPN Tools implementation. To explain them, we divide them into two categories of constraints in this paper, i.e., Tasks or relations constraints.

Tasks constraints are the unary constraints that can be used to annotate a task—shown on the right side of the figure. If a task is annotated with *init* or *last*, it shall start or end the process, respectively. The existence constraints show the occurrence of the execution of a task that is expected in a process, e.g., a task can be done once or more ($1..*$), etc. If we expect the exact occurrence of the execution of a task, we can annotate it with the exactly constraints, e.g., *exactly1* means that a task shall happen exactly once in executing a process. If we want to prohibit executing a task for a certain number of

times, we can use absence constraints, e.g., *absence2* means that the task can be executed at most once.

There are many relation constraints in Declare. Some are unidirectional, i.e., all relations having an arrow in Fig. 5 and *responded existence relation*. In the relations with an arrow, the task connected to the arrow is the latter task. In *responded existence relation*, the task connected to the line (without the bullet) is the latter one. Other relations are bidirectional.

The main three types of relations are *response*, *precedence*, and *succession*, as shown on the left side of the figure. *response* relation means that the latter task shall eventually be executed after the former one. *precedence* relation means that the former task shall be executed before the latter one. *succession* relation means that both *response* and *precedence* constraints shall hold.

These three relations can have special types called *alternate* and *chain*. In *alternate response*, the former task shall not be executed again until the latter is executed. In *alternate precedence*, the latter task shall not be executed again unless the former gets executed. *alternate succession* relation means that both *alternate response* and *alternate precedence* constraints shall hold. In *chain response*, the latter task shall be executed right after the former one. In *chain precedence*, the latter task can only be executed right after the former one. *chain succession* relation means that both *chain response* and *chain precedence* constraints shall hold.

The *not co-existence* relation means either former or latter task can be executed. The *not secession* relation means that the latter task cannot be executed after the execution of the former one. The *not chain secession* relation means that the latter task cannot be executed right after the execution of the former task.

The *responded existence* relation means the latter task shall be executed in the process if the former one gets executed. The latter task can be executed before or after the former one. The *co-existence* relation means that if one of the tasks gets executed, the other one shall also be executed in the process. The *choice* relation means that one of the related tasks shall be executed. Note that both tasks can also be executed. The *exclusive choice* relation means that only

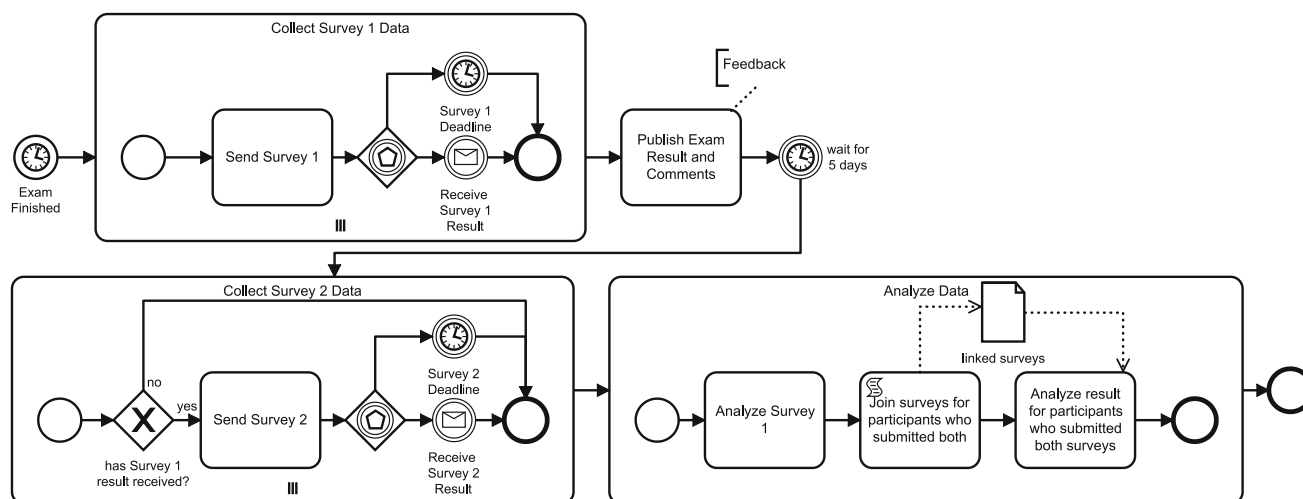


Fig. 6 The research steps

one of the related tasks shall be executed, so both tasks cannot be executed. For detail, we refer readers to [35].

3 Research method

Davis F.D. describes how the actual system's usage depends on the attitude of users, which can be predicted using two variables: perceived usefulness (PU) and perceived ease of use (PEU) of a system [13]. He also defined measurement scales based on which these variables can be evaluated [14]. These measures are widely used to evaluate different information systems, including various business process modeling languages, e.g. [27, 29, 31, 32, 44].

This paper adopted the technology acceptance model to evaluate how users perceive DCR, CMMN, and Declare using these variables. The user acceptance evaluation is a subjective score because users need to respond based on self-assessment. Thus, the result can vary during repeated experiences due to self-assessment biases. These biases are rooted in over- or under-confidence, which can be minimized by repeated experience and feedback [17]. Indeed, the answers will be more reliable when these biases are minimized.

To minimize the over- and under-confidence biases, the students are trained in the business process and case management course at Stockholm University, where they practice the languages through assignments that were designed based on Experience-Based Learning and Agile Principle [26]. They designed process models using real-case scenarios for which they received feedback. They also had sessions with an external expert to ask questions about the DCR language. Then, they participated in the examination where they needed to design models for a given case description individually.

The exam's case description is used as a simple task based on which the students' perceptions are collected before and after receiving the feedback. This enables checking if there were any significant changes in students' perceptions due to the given feedback.

The data collection and data processing start after the exam, which is shown in Fig. 6. Participation in this study was voluntary, where data were collected from students through two surveys—conducted before and after giving feedback on their exams. We call them Survey 1 and Survey 2, respectively. The surveys were identical, but students did not know about it beforehand. The questions are defined based on [13], where participants could respond by choosing options in the range of *extremely unlikely* (1), *quite unlikely* (2), *slightly unlikely* (3), *neither likely nor unlikely* (4), *slightly likely* (5), *quite likely* (6), and *extremely likely* (7). In addition, students' opinions about what they have liked or disliked about each language are collected. The list of questions can be found in Appendix A.

After the exam, survey 1 is sent out, and the responses are collected upon the announced deadline. Then, the grades and comments for the examination result are published. Students had some days to go through the comments and discuss their questions with the teacher. Then, the second survey is sent out to those who participated in the first survey. The data are collected before the announced deadline.

This paper analyzes the collected data to understand how participants perceived each language. Also, it has been investigated if there is any significant difference between participants' perceptions of those languages. The internal consistency of responses is checked using Cronbach's alpha, a widely used technique to test the reliability, e.g. [8, 12, 37, 39]. The Cronbach's alpha value above 0.7 is usually considered reliable.

The second survey's responses are linked with the first one to analyze the result before and after the feedback. The collected data are analyzed by linking the data sources containing both Survey 1 and Survey 2 data. This paper includes the result of students who participated in both Survey 1 and Survey 2, enabling us to track how opinions are changed after receiving the feedback. It has been investigated if there is any significant difference in participants' perception due to receiving feedback.

This paper includes two studies. The first study focuses on assessing DCR and CMMN through the described method. The result is used to answer the first two research questions. The second study assesses DCR, CMMN, and Declare through the described method. It only includes students with working experience to make the assessment more relevant. The second study is conducted to answer the third research question.

4 Result and discussion

This section reports and discusses the results of two studies conducted in 2020 and 2022 to answer research questions. In these studies, we invited students from the Business Process and Case Management course, which is a part of the "Master's Programme in Open eGovernment" at Stockholm University. The program and its courses are offered in distance mode, and it is usual to have students who already work in the industry at the same time in this course. In 2020, students learned about BPMN in addition to DCR and CMMN. However, BPMN has been replaced with Declare from 2021. Also, the first study includes students regardless of industrial experience; the second study only includes students with industrial working experience. The details are given below separately.

4.1 Study 1

This study is designed to answer the first and second research questions. This part contains several subsections presenting (i) an overview of participants, (ii) how participants perceived CMMN compared to DCR, (iii) how feedback has changed participants' perceptions, and (iv) discussion on threats to validity for this study.

Figure 7 shows the age distribution of students who participated in the final examination and the two surveys. The course curriculum included BPMN, CMMN, DCR, and Process Mining modules in 2020, so students were familiar with BPMN. However, they did not declare any prior experience with a KiP modeling language. The following tools were used for each module in this course. For BPMN and CMMN, the

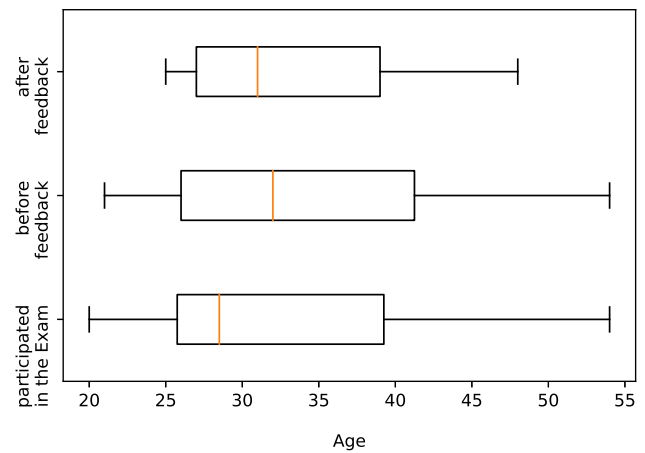


Fig. 7 The age distribution of students

online editors provided by bpmn.io^{1 2} were used. For DCR, DCR Graph³ was used. For Process Mining, Apromore⁴ was used.

Among 24 master-level students who registered for the exam, 20 students participated in *Survey 1* among which 13 students also participated in *Survey 2*. In *Survey 1*, 9 and 11 students were male and female, respectively, while in *Survey 2*, 5 and 8 students were male and female, respectively.

The average age for students who participated in the exam, *Survey 1* (before receiving feedback), and *Survey 2* (after receiving feedback) were 32.5, 34, and 34, respectively. The age distribution of students who participated in the examination is extracted from the official examination page at Stockholm University.

4.1.1 The overall perception

The left side of Fig. 8 shows an overall picture of how participants perceived the usefulness (PU) and perceived ease of use (PEU) of both DCR and CMMN languages. The median for PU and PEU is around 5 (out of 7). The first quartiles (Q1) are 4.2 and 3.7 for PU and PEU, respectively. This indicates that 75 percent of respondents rated perceived measures above 50 percent of possible value, i.e., 3.5.

4.1.2 Perceptions per language

The right side of Fig. 8 shows how participants perceived the usefulness (PU) and perceived ease of use (PEU) for DCR and CMMN languages in detail. The medians are 4.92, 5.17, 4.92, and 4.58, and the first quartiles (Q1) are

¹ <https://demo.bpmn.io/bpmn/>

² <https://demo.bpmn.io/cmmn/>

³ <https://www.dcrgraphs.net/>

⁴ <https://apromore.com/>

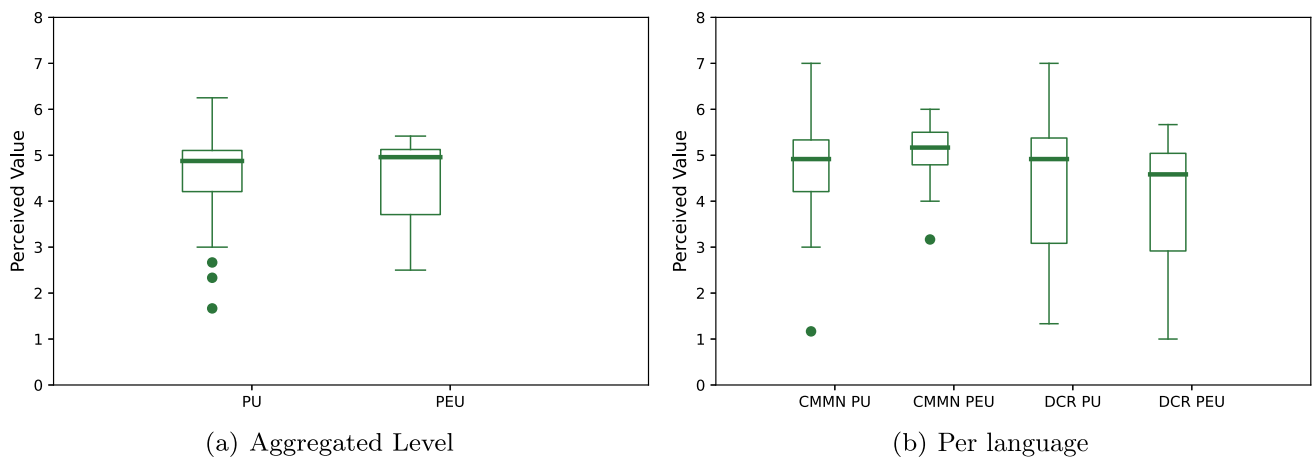


Fig. 8 The perceived usefulness and ease of use

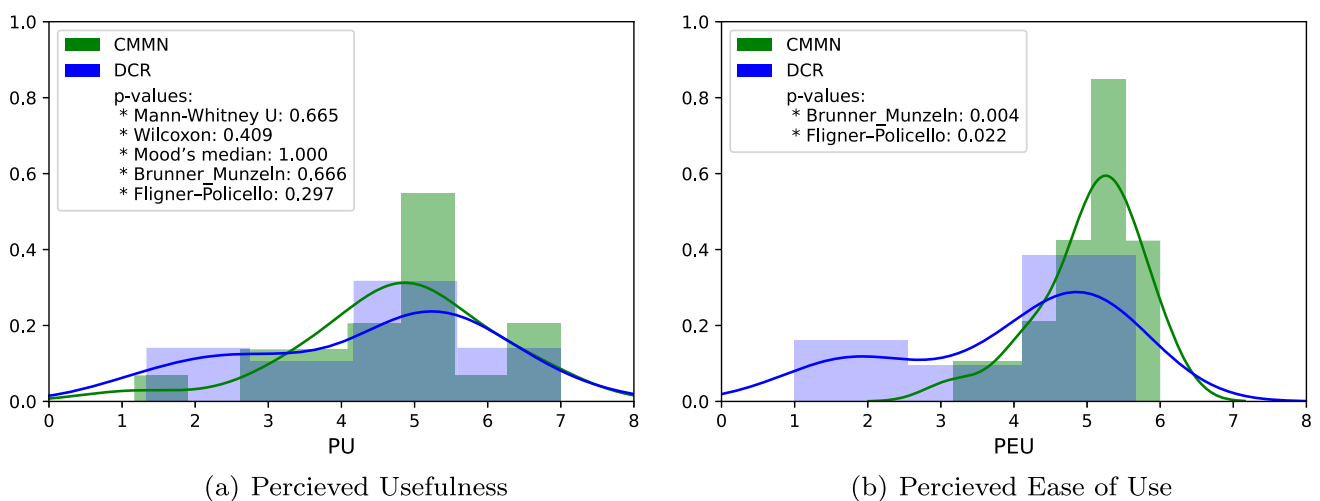


Fig. 9 Significant test for participants' perceptions of CMMN and DCR

4.21, 4.79, 3.08, and 2.92 for CMMN's perceived usefulness, CMMN's perceived ease of use, DCR's perceived usefulness, and DCR's perceived ease of use, respectively. Here, 75 percent of respondents rated perceived measures for CMMN above 50 percent of possible value, i.e., 3.5. We will investigate if these differences are significant in the next section.

4.1.3 The perception analysis

This section shows the analysis result of investigating any significant difference between how participants perceived the usefulness and ease of use of DCR and CMMN languages.

The left side of Fig. 9 shows the distribution of responses on how participants perceived CMMN and DCR to be useful. As can be seen, the data are not normally distributed, so we cannot perform t-test to identify whether there is any significant difference in responses.

If the population distributions have the same shape, we can use nonparametric statistical significance tests like Mann–

Whitney U, Wilcoxon signed-rank, and Mood's median tests. Otherwise, Brunner–Munzel and the Fligner–Pollicello tests are more appropriate. In our case, the data do not have the same distribution, but they do not significantly differ as well. Thus, we apply all of the above distribution tests to calculate the p-values.

The left side of Fig. 9 shows the p-values based on these tests. The p-values are greater than 0.05 in all tests, so we cannot reject the idea that participants perceived the usefulness of CMMN and DCR the same. Thus, there is no significant difference in how participants perceived the usefulness (PU) of DCR and CMMN languages.

The right side of Fig. 9 shows the distribution of responses on how participants perceived CMMN and DCR to be easy to use. The data are not normally distributed, so we cannot perform statistical tests like t-tests. Also, the population distributions have a different shape, so the following tests are not applicable: Mann–Whitney U, Wilcoxon signed-rank,

Table 2 Cronbach Alpha for Survey 1

| | Perceived Usefulness (PU) | Perceived Ease of Use (PEU) |
|------|---------------------------|-----------------------------|
| DCR | 0.97 | 0.94 |
| CMMN | 0.97 | 0.70 |

and Mood's median. Thus, Brunner–Munzel and the Fligner–Policello tests are applicable in this case.

The right side of Fig. 9 also shows the p-values based on these tests. The p-values are less than 0.05, so we can reject the idea that participants perceived the ease of use of CMMN and DCR the same. Thus, there is a significant difference in how participants perceived ease of use for CMMN and DCR. In this study, CMMN is perceived better in terms of ease of use than DCR.

4.1.4 The reliability analysis

As explained in the method section, we used Cronbach's alpha to test the reliability of responses. Table 2 shows the Cronbach's alpha result that we calculated per each variable per language, where all values are above 0.7, which is generally considered as the acceptable threshold. The Cronbach's alpha values for both languages' perceived usefulness and ease of use are above the acceptance threshold.

Now, we can check whether there is any significant difference between how participants perceived the usefulness and ease of use of DCR and CMMN languages before and after receiving the feedback.

Note that we analyzed the data based on participants who participated in both *Survey 1* and *Survey 2*. This means that the data to analyze the perceived usefulness and ease of use are a subset of data that is presented so far. Thus, there is a small difference between perceived usefulness and ease of use in comparison with presented result—as the number

of participants is different. The same applies to Cronbach Alpha's result.

4.1.5 The overall perception

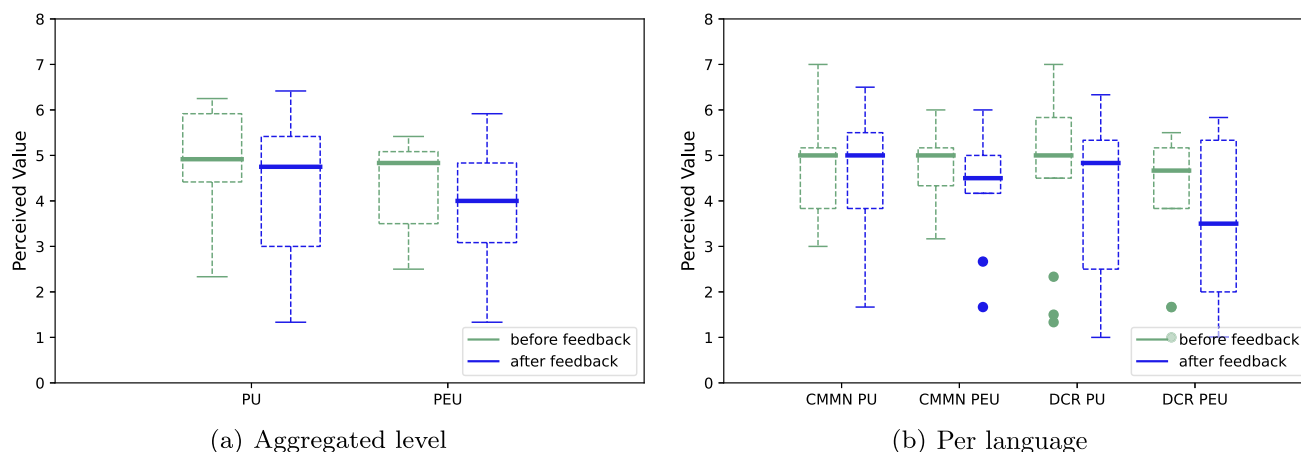
The left side of Fig. 10 shows how participants perceived the usefulness and ease of use of DCR and CMMN before and after receiving the feedback at an aggregated level. The figure is not specific for each of these languages. As can be seen in this figure, there is a slight difference between how participants reported their perceptions of perceived usefulness. The difference for Perceived ease of use is even more, where the median is lowered by one after receiving the feedback. We will check if the difference is significant later.

4.1.6 Perceptions per language

The right side of Fig. 10 shows how participants perceived the usefulness and ease of use of DCR and CMMN before and after receiving the feedback. As can be seen in this figure, there is a difference between how participants perceived usefulness after the feedback. The difference between DCR before and after the feedback seems more compared to CMMN. We will check if the difference is significant later.

4.1.7 The feedback analysis

The left and right side of Fig. 11 shows the distribution of responses on how participants perceived both languages as useful and easy to use, respectively. The p-values are greater than 0.05, so we cannot reject the idea that participants perceived the usefulness and ease of use of the two languages the same before and after receiving the feedback. Thus, there is no significant difference in how participants perceived the usefulness and the ease of use of the two languages before and after receiving the feedback.

**Fig. 10** The change in participants' perceptions

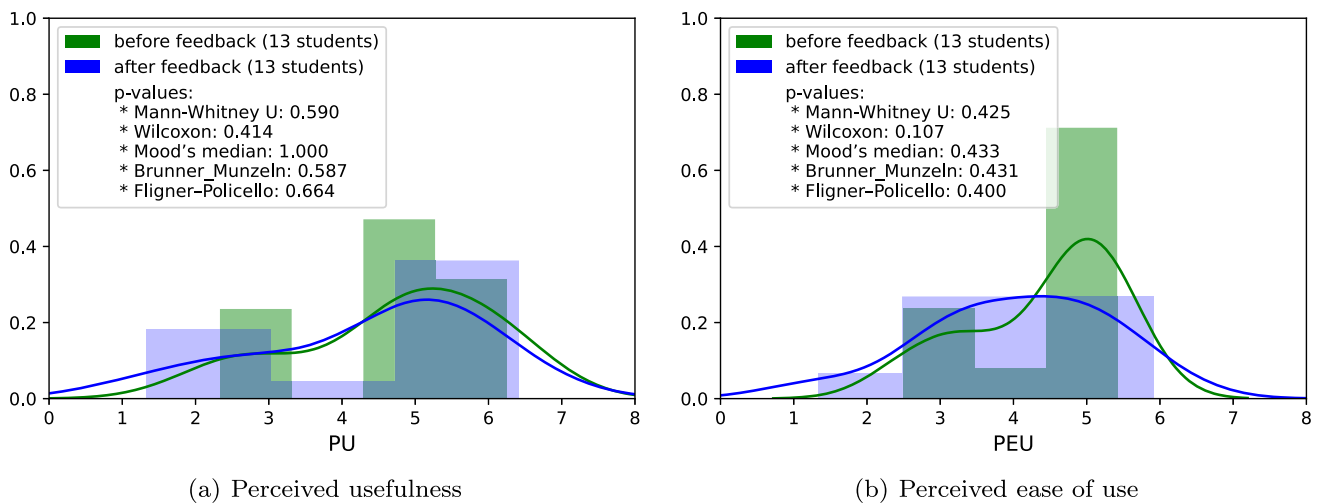


Fig. 11 Significant test of participants' perceptions before and after feedback

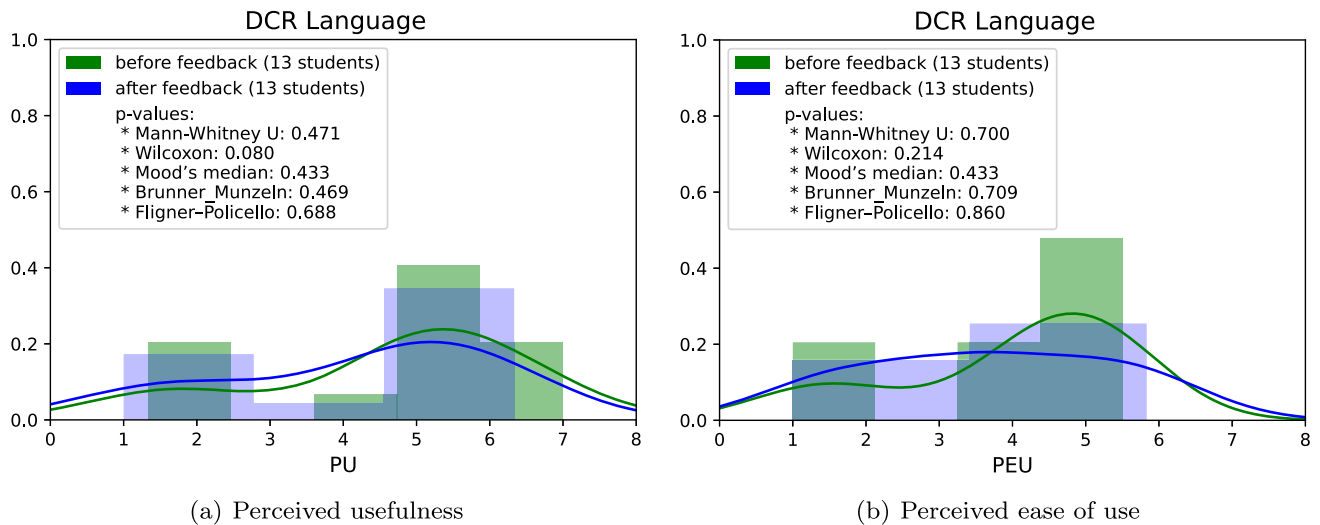


Fig. 12 Significant test of participants' perceptions for DCR before and after feedback

The left and right side of Fig. 12 shows the distribution of responses on how participants perceived DCR as useful and easy to use, respectively. The p-values are greater than 0.05, so we cannot reject the idea that participants perceived the usefulness and ease of use of DCR the same before and after receiving the feedback. Thus, there is no significant difference in how participants perceived the usefulness and the ease of use of DCR before and after receiving the feedback.

The left and right sides of Fig. 13 show the distribution of responses on how participants perceived CMMN as useful and easy to use, respectively. The p-values are greater than 0.05, so we cannot reject the idea that participants perceived the usefulness and ease of use of CMMN the same before and after receiving the feedback. Thus, there is no significant difference in how participants perceived the usefulness and the ease of use of CMMN before and after receiving the feedback.

4.1.8 The reliability analysis

Table 3 shows the Cronbach's alpha result that we calculated per variable per language before and after feedback, where all values are above 0.7. The Cronbach's alpha for all measures is quite high, i.e., above 0.9, except for the Cronbach's alpha for PEU of CMMN before the feedback, which is 0.78. Again, all values are above the accepted threshold. It is worth mentioning that Cronbach's alpha for PU and PEU for both languages before the feedback is similar to their Cronbach's alpha for the whole population reported in Table 2.

4.1.9 Confidence Interval

Figure 14 shows the means and 95% confidence interval for all measures based on which we evaluated the perceived use-

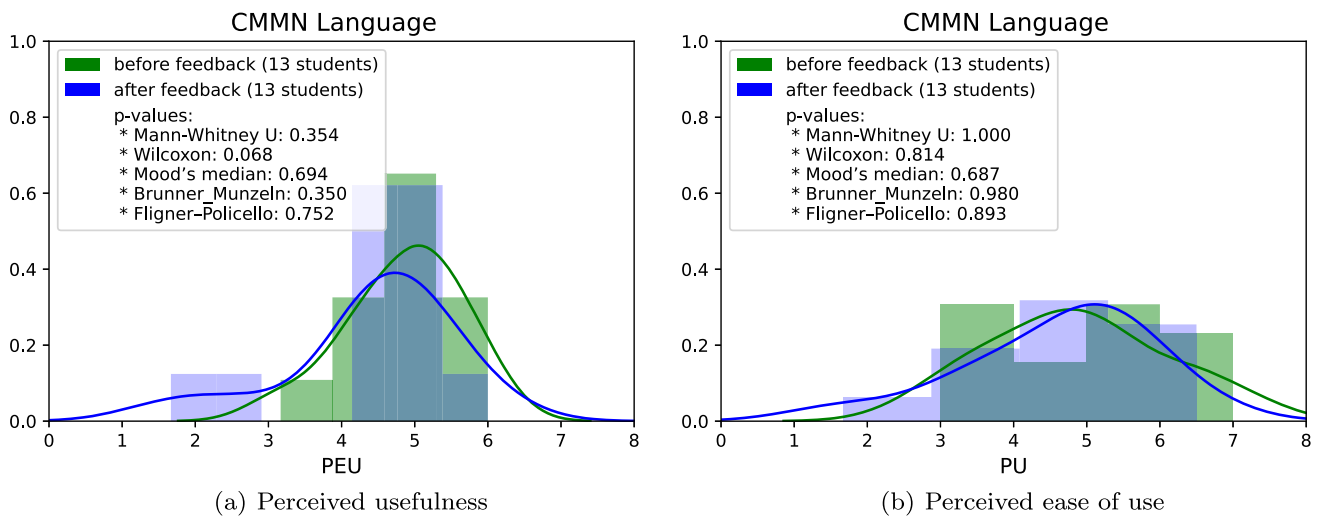
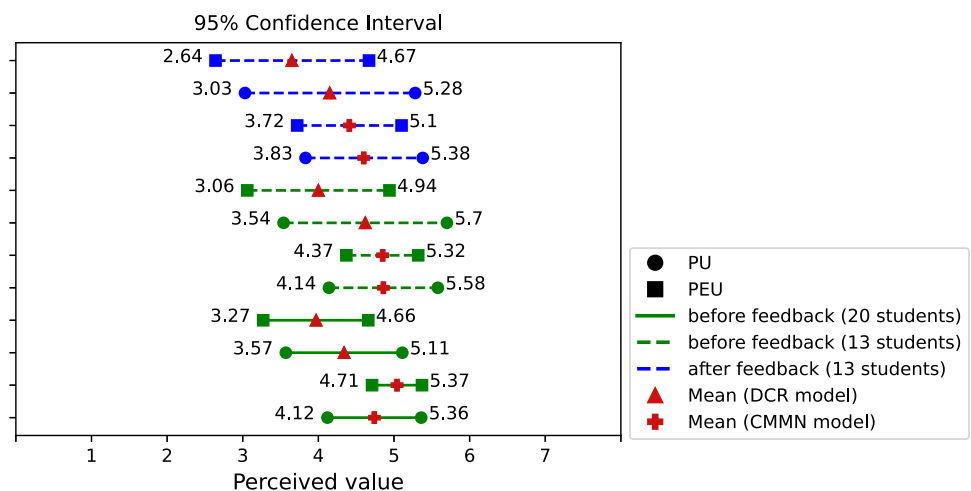


Fig. 13 Significant test of participants' perceptions for CMMN before and after feedback

Table 3 Cronbach Alpha for Survey 2

| | Perceived Usefulness (PU) | | Perceived Ease of Use (PEU) | |
|------|---------------------------|----------------|-----------------------------|----------------|
| | Before Feedback | After Feedback | Before Feedback | After Feedback |
| DCR | 0.98 | 0.98 | 0.96 | 0.97 |
| CMMN | 0.96 | 0.96 | 0.78 | 0.93 |

Fig. 14 95% Confidence Interval for the means



fulness and ease of use in this study. In this figure, the mean for CMMN and DCR models is shown by \triangle and \triangle , the perceived usefulness and perceived ease of use are shown by \bullet and \blacksquare , the participants' responses before and after the feedback is colored by green and blue, the data for Survey 1 and Survey 2 are represented by solid and dashed lines, respectively.

As can be seen in this figure, the 95% confidence interval for perceived ease of use for CMMN is between 4.71 and 5.37 (\blacksquare — \triangle — \blacksquare); while the 95% confidence interval for perceived ease of use for DCR is between 3.27 and 4.66 (\blacksquare — \triangle — \blacksquare). This is aligned with the significant test we performed to check

if these languages are perceived as significantly different in terms of ease of use.

It is also visible that the 95% confidence interval for different measures before and after feedback has not been significantly changed.

4.1.10 Limitations and threats to validity

This part reports possible limitations and threats to the validity of this study.

First, as explained and motivated in this paper, we have used students as our test subjects instead of real process designers. Students are considered valid subjects in this area

as these languages are new and are mostly unknown to practitioners outside. Thus, students can be used to evaluate how these languages can be perceived by process designers, which is also used in related work such as [4–7, 20, 43, 48, 52]. Using students as subjects can weaken the causal relation for predicting if the artifact will be used in the future. The fact that students belong to the same class and are trained under the same process can also be considered a learning bias.

Second, it shall be mentioned that students were familiar with the BPMN, which may potentially impact their PU and PEU of declarative languages. From the author's perspective, this impact is unknown, and it will be interesting to evaluate if prior knowledge of workflow-based modeling language can have a positive or negative effect! There is a significant challenge in designing such a study because imperative modeling is a dominant method in traditional business process management. Indeed, it may not be possible to find participants who are not familiar with imperative process modeling languages but are experts in process modeling.

Third, feedback can impact the subjects' opinions as they can be used as positive or negative treatment. However, the lack of feedback can result in under- or over-confidence biases. We tried our best to use neutral wording [17] to minimize this effect in this study.

Fourth, the task in the examination may be easier than real-world processes in the industry. It shall be noted that this task was one of many exercises that students had with these languages. Indeed, they have modeled process models using these languages based on real complex processes during their assignment.

Fifth, it is important to recite that the utilized tools may affect the perception of users about CMMN and DCR. These tools are means to enable users to design processes, and different features in different versions may affect how users perceive a language. It might be worth mentioning that drawing on paper is also a means when drawing a process model, which can affect the perceptions due to not having any digital assistant.

Finally, it shall be mentioned that this study only reports the result of participants' perceptions in one study. So, it is not applicable to generalize the findings to these languages, but the result may extend our understanding of knowledge-intensive processes. More experience and studies are needed to shape our understanding of these languages.

4.2 Study 2

This study aims to answer the third research question, i.e., "How do trained process designers with industrial working experience perceive the usefulness and ease of use of CMMN, DCR, and Declare compared to each other?" It follows the same method as the previous study. It shall be noted that we checked if the feedback has changed perceptions signif-

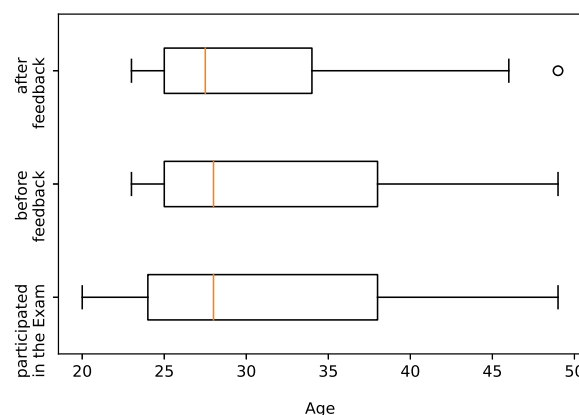


Fig. 15 The age distribution

icantly or not, and the result was aligned with the previous study. Thus, we do not repeat the same analysis for this study here.

The course had 35 students, among which 25 students participated in the exam. 17 students had working experience and participated in the first survey, among which 14 students filled out the second survey, so the study includes 14 participants, i.e., 5 females and 9 males.

Figure 15 shows the age distributions of students who participated in the examination and filled out the first and second surveys. The youngest and oldest participants had 23 and 49 years old, respectively. The mean of the participant's age was 30.8, with a standard deviation of 8.5. The minimum and maximum years of working experience were reported as 1 and 25, respectively, with a mean of 7.4 and a standard deviation of 8.5. The minimum and maximum years of IT-related working experience were reported as 0 and 24, respectively, with a mean of 3.6 and a standard deviation of 7. This indicates that participants worked more in business settings than in the IT part of their companies.

The course curriculum in 2022 included DCR, CMMN, Declare, and Process Mining, so it did not cover BPMN or any workflow-based process modeling language. For process mining, the course only focused on mining Declare and DCR process models.

One limitation of the previous study was the prior knowledge of workflow-based languages as participants were trained with BPMN in the course. As the course did not include any workflow-based languages, we asked participants about process modeling languages that they knew. There were only 4 participants who reported no prior knowledge and experience of any process modeling languages. Indeed, the majority knew one or more languages. Their answers aligned with our assumption that many people in the industry know about a workflow-based language. Thus, it will be difficult to set a setting to compare these languages

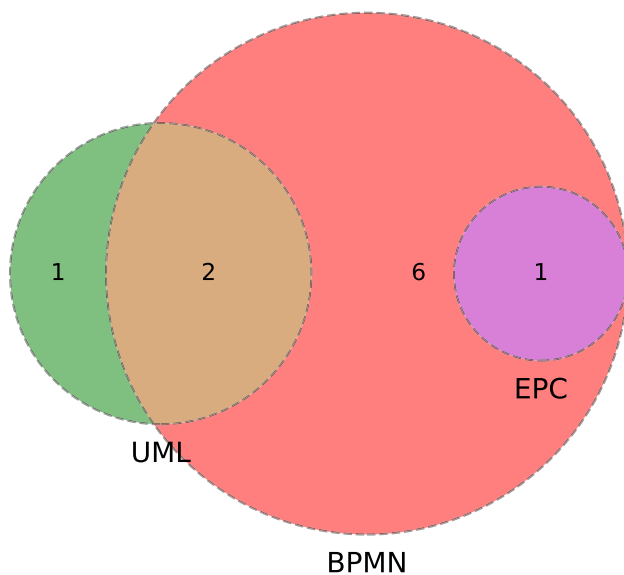


Fig. 16 Prior knowledge of business process modelling languages

with participants who do not have any prior knowledge of an imperative language but working in the industry.

Figure 16 shows the process modeling languages that participants knew or have used in practice. 9 participants declared prior knowledge of BPMN, among which 1 and 2 participants also knew EPC and UML, respectively. There was also one participant who declared prior knowledge of UML as a process modeling language.

The participants work in different countries. Figure 17 shows the geographical distribution of countries where they worked at the time of the survey. The working experience is limited to European and North American countries.

The participants reported their current roles in the organization as Business Specialist, Project Manager, Project Development Manager, Management Consultant, Technical Sales Specialist, Board Member & Tech Advisor, Project Development Manager, Exchange On-Premises Engineer & Salesforce Consultant.

The following tools were used in this course. For CMMN, the online editor provided by bpmn.io⁵ and Trisotech⁶ was used. For DCR, DCR Graph⁷ was used. For Declare, CPN Tools⁸ was used. For Process Mining, RuM⁹ and ProM¹⁰ were used.

⁵ <https://demo.bpmn.io/cmmn/>

⁶ <https://www.trisotech.com/>

⁷ <https://www.dcrgraphs.net/>

⁸ <https://cpntools.org/>

⁹ <https://rulemining.org/>

¹⁰ <https://promtools.org/>

4.2.1 The overall perception

Figure 18 shows an overall picture of how participants perceived the usefulness (PU) and perceived ease of use (PEU) of the languages used in the study. The median for PU and PEU is around 4.3 and 4.2 (out of 7), which is a little lower than the previous study. The first quartiles (Q1) are 3.9 and 3.7 for PU and PEU, respectively. This indicates that 75 percent of respondents rated perceived measures above 50 percent of possible value, i.e., 3.5, which is the same as the previous study.

4.2.2 Perceptions per language

Figure 19 shows how participants perceived the usefulness (PU) and perceived ease of use (PEU) for DCR, CMMN, and Declare languages in detail. The medians are 5.3, 4.7, 4.6, 4.3, 4, and 3.8 for these languages, respectively. The first quartiles (Q1) are 4.5, 4.1, 3.8, 3.6, 3.2, and 3.5 for these languages' PU and PEU, respectively. Here, 75 percent of respondents rated perceived measures for DCR and CMMN above 50 percent of possible value, i.e., 3.5. However, this does not hold for Declare, which has a lower median and Q1.

4.2.3 The reliability analysis

As explained in the method section, we used Cronbach's alpha to test the reliability of responses. Table 4 shows the Cronbach's alpha result that we calculated per each variable per language, where all values are above 0.7, which is generally considered as the acceptable threshold. The Cronbach's alpha values for both languages' perceived usefulness and ease of use are above the acceptance threshold.

4.2.4 Confidence interval

Figure 20 shows the means and 95% confidence interval for all measures based on which we evaluated the perceived usefulness and ease of use in this study. In this figure, the mean for DCR, CMMN, and Declare models are shown by ▲, + and ★, respectively. The perceived usefulness and perceived ease of use are shown by ● and ■. From the 95% confidence interval, it is visible that DCR is perceived better than CMMN, and CMMN is perceived better than Declare for both perceived usefulness and perceived ease of use. We will investigate whether these differences are significant or not in the next section.

4.2.5 The perception analysis

Figure 21 shows the distribution of responses on how participants perceived usefulness and ease of use for KiP languages. As can be seen, the data are not normally distributed,

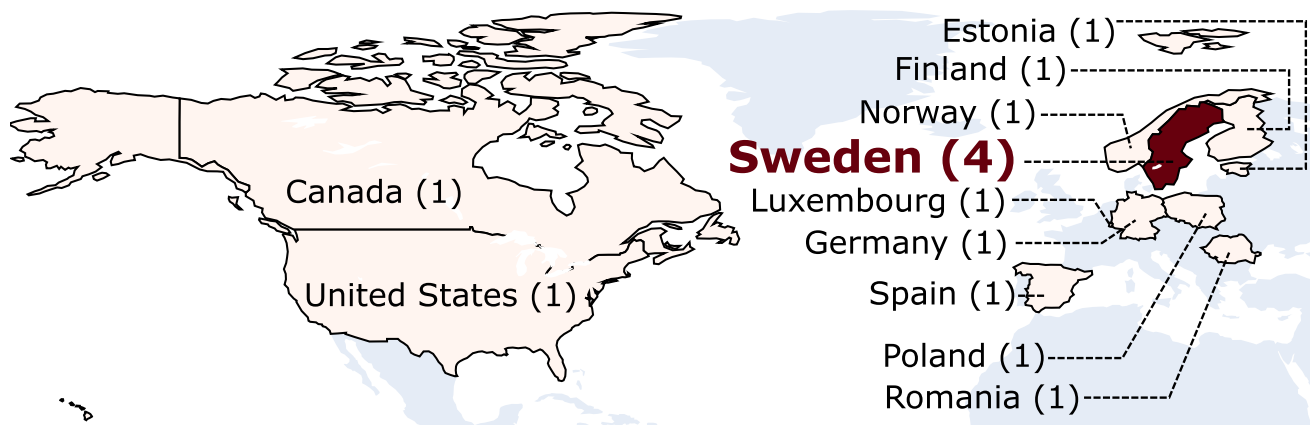


Fig. 17 The countries of participants

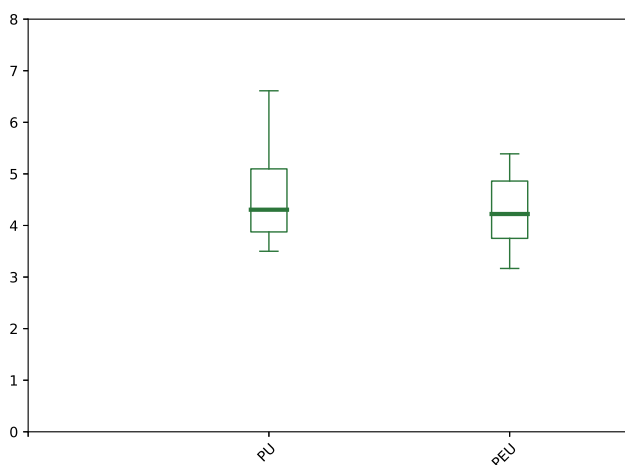


Fig. 18 The perceived usefulness and ease of use on aggregated level

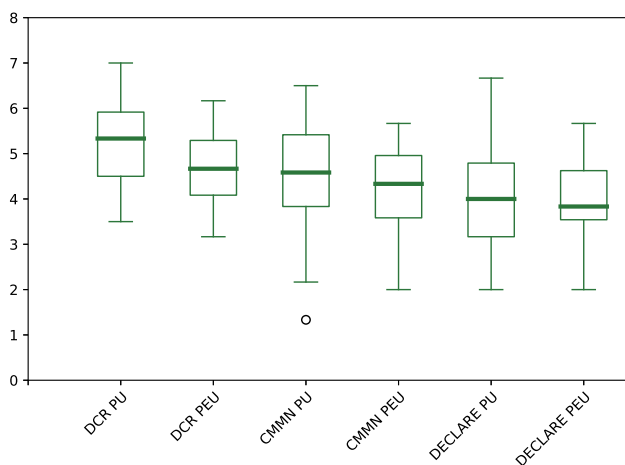


Fig. 19 The perceived usefulness and ease of use per language

Table 4 Cronbach Alpha for Survey 2

| | Perceived Usefulness (PU) | Perceived Ease of Use (PEU) |
|---------|---------------------------|-----------------------------|
| DCR | 0.89 | 0.88 |
| CMMN | 0.95 | 0.85 |
| DECLARE | 0.96 | 0.82 |

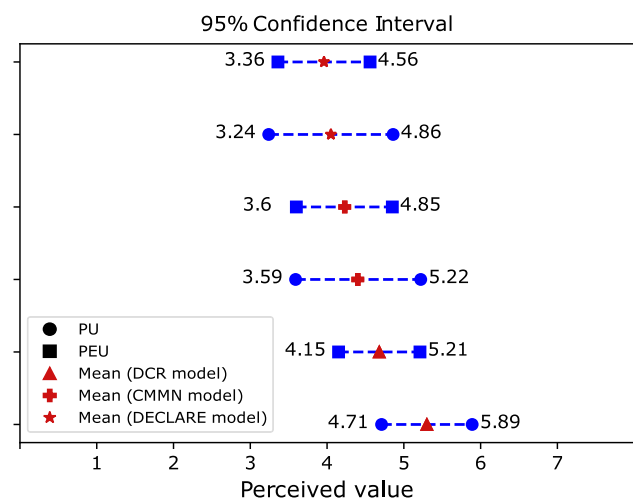


Fig. 20 95% Confidence Interval for the means

so we cannot perform t-test to check if the perceptions are significantly different.

The population distributions do not have the same shape, so we cannot use nonparametric statistical significance tests like Mann–Whitney U, Wilcoxon Signed-Rank, and Mood’s median tests. Fligner–Policello test is also not applicable as the medians are different for the perceptions of each language. Thus, Brunner–Munzel test is the only suitable option for our study. Therefore, we applied this test to calculate the p-value, which is specified for each pair of languages in the figure.

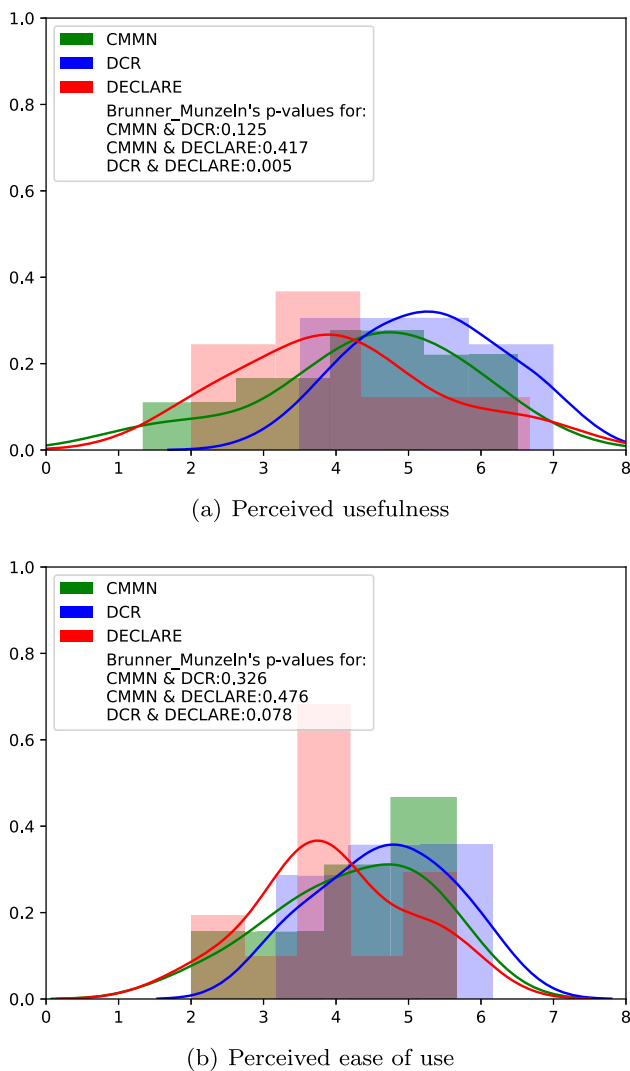


Fig. 21 Significant test of perceptions per each pair of language

The result shows that there is only one significant difference between compared languages, which is the comparison between the perceived usefulness of DCR and Declare ($p < 0.05$). Thus, participants perceived the usefulness (PU) of DCR significantly better than Declare. The other differences are not significant.

4.2.6 Limitations and threats to validity

This study has the same limitations as the previous one, except we only include participants with industrial working experience. Thus, this study can provide stronger results in measuring perceived usefulness and ease of use from participants with working experience. Although the course did not include BPMN, but most participants were familiar with at least one workflow-based process modeling language, which applies the same limitation as the previous study. As the per-

ception of CMMN and DCR are aligned with the previous study, the limitations of the previous result due to including one study are weakened, yet other limitations are still held.

4.3 Participants' feedbacks

In surveys, participants could give positive and negative feedback about each language. These feedbacks are a sort of open-ended qualitative data, which are coded by manually assigning tags to each feedback in an inductive way. As a result, we ended up with seven categories, i.e., *Collaboration Support*, *Number of Elements*, *Modularization Support*, *Simulation Support*, *Roles Modeling*, *Traceability Support*, and *Large Processes*.

Figure 22 shows the number of comments for each category for each language, where the left- and right-side figures focus on positive and negative feedback, respectively. In general, DCR and Declare received the most and the least positive feedback, respectively. Also, Declare and DCR received the most and the least negative feedback, respectively.

Some of the comments are related to the languages, but most of the comments are related to how the tools support the language. This indicates the importance of the tool in assessing the perceived usefulness and ease of use of different process modeling languages. The tools are the means through which process engineers design a business process, and they can affect how users perceive a language. It is worth re-emphasizing that using pen and paper is also a means when drawing a process model, which can affect the perceptions due to not having any digital assistant. Here, we present some of the feedback based on each category.

4.3.1 Simulation support

The participants had positive feedback about simulation support, which was the most frequent item in the feedback. They identified the simulation feature as the most important asset that helps in learning a language. The DCR simulation is appreciated by participants much more. One participant stated the importance of undoing the previous actions in an interactive simulation. (S)he reported this as a very important feature for analysis purposes as it can enable process analysts to investigate different scenarios easier—without a need to restart the simulation.

4.3.2 Traceability support

The second appreciated positive feedback was the traceability provided by DCR, a.k.a., DCR Highlighter. It enables process designers to highlight the given case description to create the process models. This tool enables traceability between model and text, which participants stated as very useful, e.g., a participant wrote this feedback: “Gen-

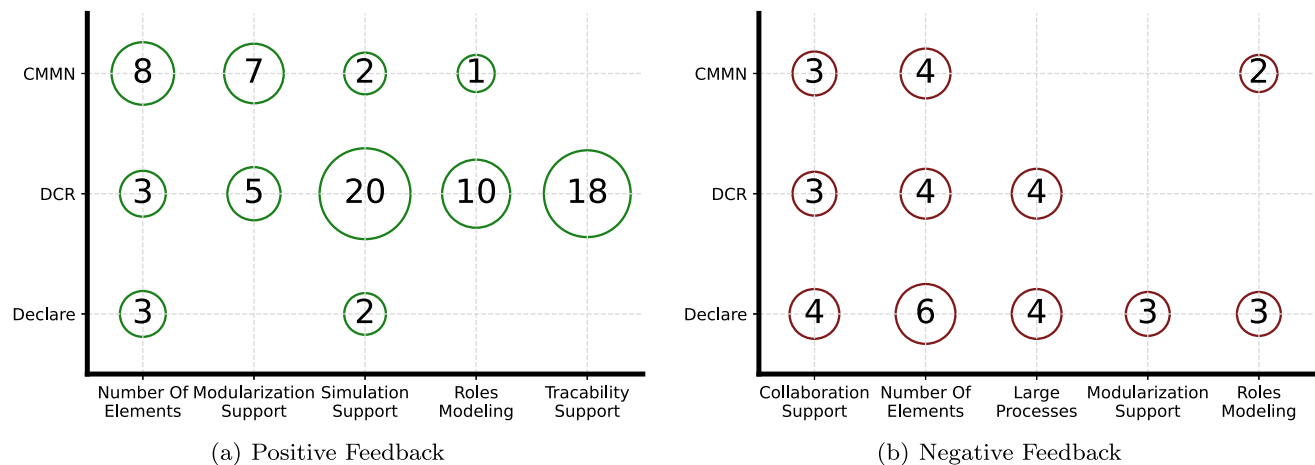


Fig. 22 Number of participants' feedback per categories

erally DCR is easy to use. I particularly like the highlighter functionality; it is very useful tool when identifying roles, activities and rules from the process description." The feedback confirms the study result published in [1].

4.3.3 Modularization support

Modularization Support had both positive and negative feedback types. CMMN and DCR received positive feedback, yet Declare received negative ones. CMMN received slightly more positive feedback due to supporting modularization using stages.

One participant also commented on the importance of discretionary items along the stages, i.e., "One useful functionality of CMMN is the Stage concept providing a way of grouping tasks and determining the context inside the case. In addition to stages the definition of potential upcoming items, called discretionary items was a powerful functionality."

For DCR, the feedback was positive, yet it was a comment on this feature as DCR has two ways to support modularization, i.e., nesting activities under activities or a process. One participant found this as a "cognitive load" in design time despite categorizing this feature as positive in general for DCR.

For Declare, we received three negative feedback about the lack of support for sub-processes in CPN Tools. It shall be mentioned that participants knew that Declare could support modularization in theory as they have read [20] as a part of the course literature based on which they wrote an assignment. However, they expressed the lack of tool support as a negative point for applications in practice, e.g., one participant wrote, "the lack of sub-process is the only complaint I have on this language. I fear bigger models can become too messy without them."

4.3.4 Roles modeling

This feature is also recognized as important, where we received positive feedback for DCR yet negative for Declare. As a participant commented for Declare, "It is not possible to combine the control flow and organizational circumstances which limits the language for implementation in practical cases."

For CMMN, we received both positive and negative feedback. One participant appreciated that the language supports role modeling through event listener and human activity, yet two participants had negative comments on this support. The use of event listener is perceived as negative as it makes the model complicated, according to one participant, i.e., "Complexity grows further whenever roles are depicted using event listeners that might hinder leanness and comprehensibility." Human activity is also reported as not useful as the name of the role is not visible in the model.

4.3.5 Collaboration support

Although the examination was individual, we received feedback from students in the questionnaire regarding the collaboration capabilities of the tools. This shall come from their experience of using the tools in the course for their projects, where they worked in teams to design process models for real cases. The main critique was the lack of capability to support multiple users to work simultaneously on a process model together, which applies to all languages.

4.3.6 Number of elements

As can be seen in Fig. 22, we received both positive and negative feedback for all languages in this category.

For CMMN, participants gave positive feedback on discretionary items, entry and exit sentry, the limited number

of notation elements, and having different shapes for graphical notations. The negative feedback was about decorators of tasks and stages that could change the semantics of these elements, especially the manual activation decorator. We will explain this issue in detail at the end of this section.

For DCR, students wrote positive feedback about the include, exclude, response, and pre-condition. The negative feedback was about spawn, non-response, and the use of identical graphical notations (with a different color) for pre-condition and milestone.

For Declare, the task constraints, including *init* and *last*, were reported as very positive. However, it was very negative comments on the number of graphical notations, e.g.,

- “[...] It might become overwhelming to learn and then to keep in mind.”
- “There are so many different constrains. I guess a typical user at work would need to have the PDF manual opened almost all the time.”

4.3.7 Large processes

This category only captures negative comments related to DCR and Declare. Most of the feedback was related to the readability of Declare and DCR when capturing large processes. Here are examples of comments:

- “some arrows are easy to understand, but they become easily complicated in a model. the big number of arrows in a real process would create a mess.” (a comment for DCR).
- “Having too many condition arrows confused me a lot and made it difficult for me to create effective modelling, so I consider the amount of condition possibilities the least useful.” (a comment for Declare).

Overall, feedbacks refer to the difficulty of capturing large processes using these languages.

4.4 Discussion on different perceptions for CMMN

In study 1, CMMN was perceived as significantly better than DCR in terms of perceived ease of use. However, this difference is not preserved in study 2. As the difference was significant in the first study, this change may be the result of changing the study setting, where: (i) BPMN was taught in the first study but not in the second one, (ii) Participants could not simulate CMMN models in the first study due to the tool's limitation, yet they could simulate them in the second study as we used Trisotech, (iii) Declare was taught in the second study but not in the first one, and (iv) Imperative Process Mining was taught in the first study in comparison to the declarative one in the second study.

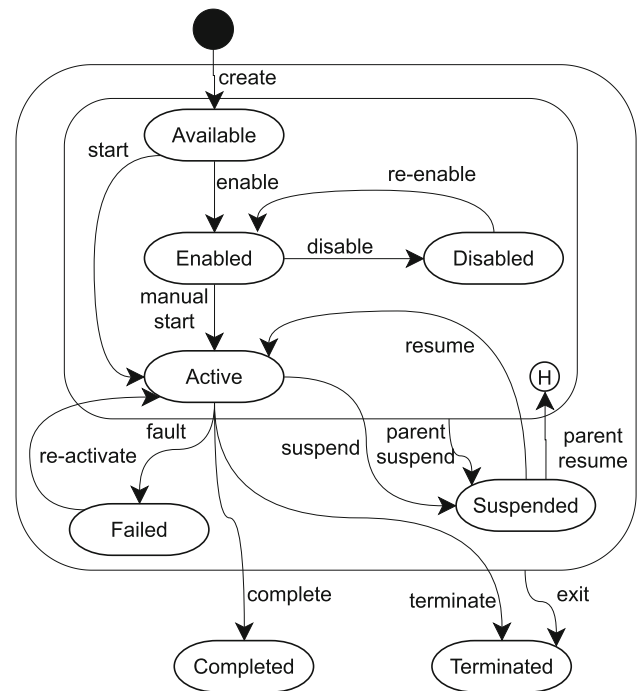


Fig. 23 Lifecycle of a Stage or Task instance in CMMN, redrawn from [38]

The changing factor cannot be known without a further study, yet it can be assumed that simulator played an important role based on discussion with participants during the course and received feedback. As reflected in feedback, decorators of tasks and stages can change the semantics of these elements. The simulator enables participants to check these semantics by designing different models, so the simulator extended their capability to understand the language further as they were not limited by feedback that they received during the course. There was one challenging situation in CMMN, which was discussed in the course that we explained below.

The Trisotech simulator enabled students to clearly identify how decorating a task or stage using a manual activation (\triangleright) can change the behavior of the model in regard to the exit sentry evaluation. To understand this issue, we can look at the stage or task instance lifecycle as described by the CMMN specification [38]. Figure 23 shows this lifecycle that has several states and transitions. The issue that we explain here applies to both stage and task. We only explain the issue for tasks through one example.

For any CMMN task, the lifecycle starts by creating an instance of the task, so the instance will be in *Available* state. As mentioned by the CMMN specification, “Entry criterion sentries are considered ready for evaluation while the task, stage, or milestone is in *Available* state” [38]. Thus, the decoration of tasks will not affect how they will be enabled because all of them meet the *Available* state at beginning, where the entry sentries (\diamond) will be evaluated.

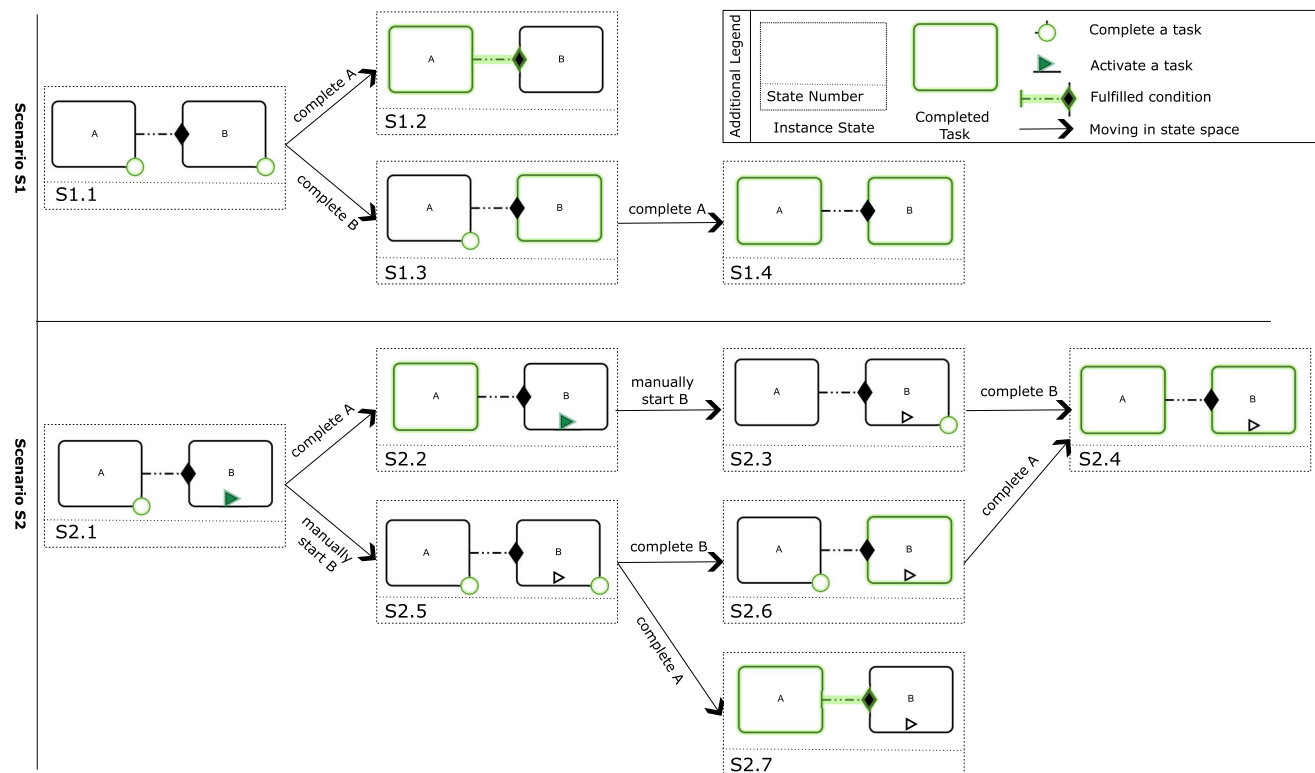


Fig. 24 State space analysis for two CMMN models showing how decorating a task that has exit sentry with a manual activation can increase the possible execution states

The change in the state of the instance after *Available* state depends on whether it is decorated with manual activation (\triangleright) or not. If a task is decorated with the manual activation decorator (\triangleright), then its instance's state will proceed to *Enabled* if it has no entry sentry—or if there are entry sentries, one of them gets evaluated as True. Then, it shall wait until a human manually starts the instance to become in active mode. However, if a task is not decorated with the manual activation decorator, its instance's state directly moves to *Active* from *Available* upon meeting the conditions explained before.

A task can also have an exit sentry (\blacklozenge). As mentioned by the CMMN specification, “Exit criterion sentries are considered ready for evaluation while the CasePlanModel, State, or Task is in Active state” [38]. This means that if a task is decorated by the manual activation decorator (\triangleright) and its instance is in the *Enabled* state, the exit criterion will not be evaluated. It might be worth citing that “Sentries are evaluated when events arrive to the system or when events are generated by the system” [38], so the system will not queue them to process later. It might be worth mentioning that the movement from other states to *Terminated* state in the lifecycle is related to the children's states—as this lifecycle is for both Tasks and Stages. We acknowledge Trisotech's R&D department for helping us to understand this behavior in detail.

This part of semantics increased the cognitive load for participants to understand how CMMN models would work in practice. To make it clear, Fig. 24 shows the state space analysis of two simple scenarios with the help of Trisotech's simulator, i.e., S1 and S2. These two scenarios contain two tasks, called A and B, where A is connected to the exit sentry (\blacklozenge) of B. The only difference is that B is decorated with manual activation (\triangleright) in Scenario S2. In this figure, the instance's states are represented by a dashed rectangle where the instance numbers are written in their bellow sections. The action that causes a change in state space is shown by an arrow. The green circle at the bottom right of tasks shows that they are in *Active* state, so the user can complete them. A task with a green border shows that the instance of the task is in *Completed* state. A green manual activation icon shows that a task is in *Enabled* state. A green relation between two tasks containing an exit sentry shows that the exit sentry criterion is evaluated. The picture for each state is taken from the simulation of these cases from Trisotech Case Modeler tool.

In *Scenario S1*, the instances of tasks A and B are in *Active* state at the beginning as shown by S1.1 state. If the user completes the instance of task A, the instance of task A's state will change to *Completed*, and the exit sentry of the instance of task B will be evaluated, which terminates the

instance of task *B*—as shown by *S1.2*. On the other hand, if the instance of task *B* gets completed (state *S1.3*), the user can still complete the instance of task *A*—as shown in *S1.4*.

In *Scenario S2*, the instance of task *A* is in *Active* state but the instance of task *B* is in *Enabled* state at the beginning—as shown by *S2.1* state. The reason is that task *B* is decorated with manual activation (\triangleright). If the user completes the instance of task *A*, the instance of task *B* will not be terminated as it is not in *Active* state, so the state space move to *S2.2*. At this state, the user can manually start the instance of task *B*, moving the state space to *S2.3*, and (s)he can complete the instance of task *B*, which moves the state to *S2.4*.

In this scenario, the user can manually activate the instance of task *B* in *S2.1* instead, which moves the state space to *S2.5*. In state *S2.5*, if the user completes the instance of task *A*, the instance of task *B* will be terminated—as shown by *S2.7*. In state *S2.5*, if the user completes the instance of task *B*, the state space moves to *S2.6*, where the user can still complete the instance of task *A*. Completing the instance of task *A* moves the state space to *S2.4*.

We can summarize these two scenarios accordingly:

- In *Scenario S1*, if *A* gets completed first, *B* cannot get completed.
- In *Scenario S2*, if *B* is manually started first AND *A* gets completed then, *B* cannot get completed.

The state space analysis shows how annotating a task with a manual decoration can complicate the modeling as users need to consider two conditions to think about the model's outcome. It shall be mentioned that there is a way to overcome this complexity in CMMN. In *Scenario S2*, if the process designer encapsulates task *B* within a stage and moves the exit sentry to the boundary of the stage rather than task *B*, then the state space will be similar to *S1*, yet this needs extra design elements in the model.

In summary, we do not know if the participants in Study 1 were completely aware of such complexity in CMMN, but the participants in Study 2 knew it due to the possibility of using simulation in the course. This fact shows a future direction for doing research on the role of using tools with simulation capability in evaluating modeling languages. Also, we could not find any CMMN model in related work reporting the use of manual activation, which can affect how users perceived this language in other studies.

5 Conclusion

This paper reports on how students perceived knowledge-intensive business process modeling languages in terms of usefulness and ease of use. The paper includes two studies conducted in 2020 and 2022, where the first one focused on

CMMN and DCR, while the second one included Declare as well. The studies were performed by applying the technology acceptance model, where master-level students were educated on these languages, and feedback was given to reduce perception biases. The second study only includes students with working experience to investigate the perceptions of people with an industrial working background. The participants' perceptions are collected through two surveys, one before and one after feedback on their final practice in the exam.

The participants' perceptions changed a little before and after receiving the feedback. Three nonparametric statistical significance tests were performed, and the results indicate that the feedback did not significantly change the students' perceptions. The reliability of responses was also evaluated using Cronbach's alpha, which showed an acceptable level of reliability in the students' responses.

In Study 1, CMMN was perceived significantly better in terms of ease of use. In Study 2, DCR was perceived significantly better than Declare in terms of perceived usefulness. The comparison of results on users' perceptions indicates potential reasons for differences, including the importance of supporting interactive simulation by the tool when learning a language. The tool's simulation enabled participants in the second study to point out an issue they found hindering usability. Several requirements were also compiled from the feedback received from participants to improve the usability of these modeling languages.

Future directions include the need to investigate how users perceive workflow-based business process modeling languages through different study setups. It would be interesting to explore how users' prior knowledge of such languages can influence their perceptions when learning declarative modeling languages. Additionally, it would be worthwhile to investigate how extra features in modeling toolsets can improve the perceived usefulness and ease of use of languages in general. Another important area of research is to examine how the presence or absence of interactive simulation for users can affect their perception of a modeling language. Finally, a promising avenue of research is to study how different teaching methods, including feedback, can impact student learning outcomes, which can be done, e.g., by applying data-based analysis methods such as process mining to educational data.

Acknowledgements We appreciate all support that is provided by Morten Marquard from dcrgraphs.net, without which it was difficult to train the students and perform this study. We also acknowledge Trisotech by providing us with their toolset to enable students to model and simulate CMMN models. Also, we appreciate their feedback and help in understanding CMMN specification better.

Funding Open access funding provided by Stockholm University.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

Appendix A: Questionnaire

This section documents the questions for measuring the perceived usefulness and ease of use in the survey.

Appendix A.1: Structure of the survey

In the survey, we asked about the gender and age of participants. In addition, each survey had a unique identifier helping us to match the collected responses before and after receiving feedback. For Study 2, we also asked questions about working experience, the country and etc. in addition to questions asked in Study 1.

Appendix: A.1.1 Questions about perceptions

The questions to measure perceived usefulness and ease of use of each language have been designed based on the technology acceptance model [14]. Students could choose answers based on scales from *extremely unlikely* to *extremely likely*:

1. extremely unlikely
2. quite unlikely
3. slightly unlikely
4. neither likely nor unlikely
5. slightly likely
6. quite likely
7. extremely likely

These are the questions used to measure the *perceived usefulness* of each language. [language_code] is set as 1,2 and 3 for DCR, CMMN and Declare, respectively. [language_name] refers to the name of the language asked in the survey.

- S.[language_code].1.1: Using [language_name] in this experience enabled me to model business processes more quickly.

- S.[language_code].1.2: Using [language_name] in this experience improved my performance when modeling business processes.
- S.[language_code].1.3: Using [language_name] in this experience increased my productivity when modeling business processes.
- S.[language_code].1.4: Using [language_name] in this experience enhanced my effectiveness in modeling business processes.
- S.[language_code].1.5: Using [language_name] in this experience made it easier for me to model business processes.
- S.[language_code].1.6: I found [language_name] useful in this experience.

These are the questions used to measure the *perceived ease of use* of each language:

- S.[language_code].2.1: Learning how to use [language_name] was easy for me.
- S.[language_code].2.2: I found it easy to get [language_name] to do what I wanted to do.
- S.[language_code].2.3: I found [language_name] understandable and clear.
- S.[language_code].2.4: I found [language_name] flexible to be used in modeling business processes.
- S.[language_code].2.5: It was easy for me to become skillful at using [language_name].
- S.[language_code].2.6: I found [language_name] easy to use.

Appendix: A.1.2 Questions about participants' comments

These are the questions used to collect participants' comments about both languages:

- S.[language_code].3.1: What did you like about [language_name]?
- S.[language_code].3.2: What did you dislike about [language_name]?

References

1. Andaloussi, A. Abbad, Buch-Lorensen, J., López, H.A., Slaats, T., Weber, B.: Exploring the modeling of declarative processes using a hybrid approach. In International Conference on Conceptual Modeling, pages 162–170. Springer, (2019)
2. Andaloussi, A. Abbad., Zerbato, F., Burattin, A., Slaats, T., Hildebrandt, T.T., Weber, B.: Exploring how users engage with hybrid process artifacts based on declarative process models: a behavioral analysis based on eye-tracking and think-aloud. Software and Systems Modeling **20**(5), 1437–1464 (2021)

3. Alman, A., Di Ciccio, C., Haas, D., Maggi, F.M., Nolte, A.: Rule mining with rum. In 2020 2nd International Conference on Process Mining (ICPM), pages 121–128. IEEE, (2020)
4. Andaloussi, A.A., Buch-Lorensen, J., López, H.A., Slaats, T., Weber, B.: Exploring the modeling of declarative processes using a hybrid approach. In ER Conference, pages 162–170. Springer, (2019)
5. Andaloussi, A.A., Burattin, A., Slaats, T., Petersen, A.C.M., Hildebrandt, T.T., Weber, B.: Exploring the understandability of a hybrid process design artifact based on dcr graphs. In Enterprise, Business-Process and Information Systems Modeling, pages 69–84. Springer, (2019)
6. Andaloussi, A.A., Davis, C.J., Burattin, A., López, H.A., Slaats, T., Weber, B.: Understanding quality in declarative process modeling through the mental models of experts. In Business Process Management Conference, pages 417–434. Springer, (2020)
7. Andaloussi, A.A., Slaats, T., Burattin, A., Hildebrandt, T.T., Weber, B.: Evaluating the understandability of hybrid process model representations using eye tracking: First insights. In BPM Conference, pages 475–481. Springer, (2018)
8. Arachchilage, N.A.G., Love, S.: A game design framework for avoiding phishing attacks. *Computers in Human Behavior* **29**(3), 706–714 (2013)
9. Bider, I., Jalali, A.: Limiting variety by standardizing and controlling knowledge intensive processes. In 2016 IEEE 20th International Enterprise Distributed Object Computing Workshop (EDOCW), pages 1–9. IEEE, (2016)
10. Bose, R., Maggi, F.M., van der Aalst, W.M.: Enhancing declare maps based on event correlations. In Business process management, pages 97–112. Springer, (2013)
11. Burattin, A., Maggi, F.M., Sperduti, A.: Conformance checking based on multi-perspective declarative process models. *Expert systems with applications* **65**, 194–211 (2016)
12. Chae, M., Kim, J., Kim, H., Ryu, H.: Information quality for mobile internet services: A theoretical model with empirical validation. *Electronic markets* **12**(1), 38–46 (2002)
13. Davis, F.D.: A technology acceptance model for empirically testing new end-user information systems: Theory and results. PhD thesis, Massachusetts Institute of Technology, (1985)
14. Davis, F.D. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS quarterly*, pages 319–340, (1989)
15. Debois, S., López, H.A., Slaats, T., Andaloussi, A.A., Hildebrandt, T.T.: Chain of events: modular process models for the law. In International Conference on Integrated Formal Methods, pages 368–386. Springer, (2020)
16. Di Ciccio, C., Marrella, A., Russo, A.: Knowledge-intensive processes: characteristics, requirements and analysis of contemporary approaches. *Journal on Data Semantics* **4**(1), 29–57 (2015)
17. Eberlein, M., Ludwig, S., Nafziger, J.: The effects of feedback on self-assessment. *Bulletin of Economic Research* **63**(2), 177–199 (2011)
18. Fahland, D., Lübke, D., Mendling, J., Reijers, H., Weber, B., Weidlich, M., Zugal, S.: Declarative versus imperative process modeling languages: The issue of understandability. In Enterprise, Business-Process and Information Systems Modeling, pages 353–366. Springer, (2009)
19. Fahland, D., Mendling, J., Reijers, H.A., Weber, B., Weidlich, M., Zugal, S.: Declarative versus imperative process modeling languages: The issue of maintainability. In BPM Conference, pages 477–488. Springer, (2009)
20. Haisjackl, C., Barba, I., Zugal, S., Soffer, P., Hadar, I., Reichert, M., Pinggera, J., Weber, B.: Understanding declare models: strategies, pitfalls, empirical results. *Software & Systems Modeling* **15**(2), 325–352 (2016)
21. Hildebrandt, T., Mukkamala, R.R.: Distributed dynamic condition response structures. In PLACES Workshop, (2010)
22. Hildebrandt, T., Mukkamala, R.R., Slaats, T.: Nested dynamic condition response graphs. In fundamentals of software engineering conference, pages 343–350. Springer, (2011)
23. Hildebrandt, T.T., Mukkamala, R.R.: Declarative event-based workflow as distributed dynamic condition response graphs. *arXiv preprint arXiv:1110.4161*, (2011)
24. Hull, R., Damaggio, E., De Masellis, R., Fournier, F., Gupta, M., Heath III, F.T., Hobson, S., Linehan, M., Maradugu, S., Nigam, A., et al.: Business artifacts with guard-stage-milestone lifecycles: managing artifact interactions with conditions and events. In Distributed event-based system conference, pages 51–62, (2011)
25. Hull, R., Damaggio, E., Fournier, F., Gupta, M., Heath, F.T., Hobson, S., Linehan, M., Maradugu, S., Nigam, A., Sukaviriya, P., et al.: Introducing the guard-stage-milestone approach for specifying business entity lifecycles. In Web Services and Formal Methods Workshop, pages 1–24. Springer, (2010)
26. Jalali, A.: Teaching business process development through experience-based learning and agile principle. In Perspectives in Business Informatics Research: 17th International Conference, BIR 2018, Stockholm, Sweden, September 24–26, 2018, Proceedings 17, pages 250–265. Springer, (2018)
27. Jalali, A.: Weaving of aspects in business process management. *Complex Systems Informatics and Modeling Quarterly* **15**, 24–44 (2018)
28. Jalali, A.: Evaluating perceived usefulness and ease of use of cmmn and dcr. In: Augusto, A., Gill, A., Nurcan, S., Reinhartz-Berger, I., Schmidt, R., Zdravkovic, J. (eds.) Enterprise. Business-Process and Information Systems Modeling, pp. 147–162. Springer International Publishing, Cham (2021)
29. Jalali, A., Maggi, F.M., Reijers, H.A.: A hybrid approach for aspect-oriented business process modeling. *Journal of Software: Evolution and process* **30**(8), e1931 (2018)
30. Jošt, G., Huber, J., Heričko, M., Polančič, G.: Improving cognitive effectiveness of business process diagrams with opacity-driven graphical highlights. *Decision Support Systems* **103**, 58–69 (2017)
31. La Rosa, M., Ter Hofstede, A.H., Wohed, P., Reijers, H.A., Mendling, J., Van der Aalst, W.M.: Managing process model complexity via concrete syntax modifications. *IEEE Transactions on Industrial Informatics* **7**(2), 255–265 (2011)
32. La Rosa, M., Wohed, P., Mendling, J., Ter Hofstede, A.H., Reijers, H.A., van der Aalst, W.M.: Managing process model complexity via abstract syntax modifications. *IEEE Transactions on Industrial Informatics* **7**(4), 614–629 (2011)
33. Maggi, F.M.: Declarative process mining with the declare component of prom. *BPM (Demos)*, 1021, (2013)
34. Maggi, F.M., Bose, R., van der Aalst, W.M.: Efficient discovery of understandable declarative process models from event logs. In International Conference on Advanced Information Systems Engineering, pages 270–285. Springer, (2012)
35. Maggi, F.M., Bose, R., van der Aalst, W.M.: A knowledge-based integrated approach for discovering and repairing declare maps. In International Conference on Advanced Information Systems Engineering, pages 433–448. Springer, (2013)
36. Marquard, M., Shahzad, M., Slaats, T.: Web-based modelling and collaborative simulation of declarative processes. In International Conference on Business Process Management, pages 209–225. Springer, (2015)
37. Masrom, M.: Technology acceptance model and e-learning. *Technology* **21**(24), 81 (2007)
38. Object Management Group (OMG®). Case Management Model and Notation (CMMN™)
39. Palm, J.-M., Colombet, I., Sicotte, C., Degoulet, P.: Determinants of user satisfaction with a clinical information system. In AMIA

- Annual Symposium Proceedings, volume 2006, page 614. American Medical Informatics Association, (2006)
40. Pesic, M.: Constraint-based workflow management systems: shifting control to users. (2008)
 41. Pesic, M., Schonenberg, H., Aalst, W.v.d.: The declare service. In *Modern Business Process Automation*, pages 327–343. Springer, (2010)
 42. Pesic, M., Schonenberg, H., Van der Aalst, W.M.: Declare: Full support for loosely-structured processes. In *EDOC*, pages 287–287. IEEE, (2007)
 43. Pichler, P., Weber, B., Zugal, S., Pinggera, J., Mendling, J., Reijers, H.A.: Imperative versus declarative process modeling languages: An empirical investigation. In *BPM Conference*, pages 383–394. Springer, (2011)
 44. Reijers, H.A., Slaats, T., Stahl, C.: Declarative modeling—an academic dream or the future for bpm? In *BPM Conference*, pages 307–322. Springer, (2013)
 45. Routis, I., Bardaki, C., Dede, G., Nikolaidou, M., Kamalakis, T., Anagnostopoulos, D.: Cmmn evaluation: the modelers' perceptions of the main notation elements. *Software and Systems Modeling* **20**(6), 2089–2109 (2021)
 46. Routis, I., Nikolaidou, M., Anagnostopoulos, D.: Modeling collaborative processes with cmmn: success or failure? an experience report. In *Enterprise, Business-Process and Information Systems Modeling*, pages 199–210. Springer, (2018)
 47. Routis, I., Nikolaidou, M., Anagnostopoulos, D.: Empirical evaluation of cmmn models: a collaborative process case study. *Software and Systems Modeling* **19**(6), 1395–1413 (2020)
 48. Sanchez-Ferreres, J., Delicado, L., Andaloussi, A.A., Burattin, A., Calderon-Ruiz, G., Weber, B., Carmona, J., Padró, L.: Supporting the process of learning and teaching process models. *IEEE Transactions on Learning Technologies* **13**(3), 552–566 (2020)
 49. Slaats, T.: Declarative and hybrid process discovery: Recent advances and open challenges. *Journal on Data Semantics* **9**(1), 3–20 (2020)
 50. Turetken, O., Dikici, A., Vanderfeesten, I., Rompen, T., Demirors, O.: The influence of using collapsed sub-processes and groups on the understandability of business process models. *Business & Information Systems Engineering*, pages 1–21, (2019)
 51. Van Der Aalst, W.M., Pesic, M.: Decserflow: Towards a truly declarative service flow language. In *Web Services and Formal Methods Workshop*, pages 1–23. Springer, (2006)
 52. Weber, B., Reijers, H.A., Zugal, S., Wild, W.: The declarative approach to business process execution: An empirical test. In *CAiSE*, pages 470–485. Springer, (2009)
 53. Westergaard, M., Slaats, T.: Cpn tools 4: A process modeling tool combining declarative and imperative paradigms. In *BPM (Demos)*, (2013)
 54. Zugal, S., Soffer, P., Haisjackl, C., Pinggera, J., Reichert, M., Weber, B.: Investigating expressiveness and understandability of hierarchy in declarative business process models. *Software & Systems Modeling* **14**(3), 1081–1103 (2015)

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Amin Jalali is an Associate Professor in computer and systems sciences at Stockholm University, Sweden, who obtained his Ph.D. from the same department in 2016. His research interests are business process management, business process modeling, and process mining, where he is involved in three national projects to improve health-care processes in Stockholm. He also has been on a program committee for more than ten conferences and workshops. Currently, he is responsible for three master-level courses, i.e., business process design and Intelligence, business process and case management, and data warehousing at Stockholm University. He also has extensive industrial working experience in different domains, including healthcare, finance, and the public sector, where he worked as a programmer, system analyst and designer, business intelligence developer, data engineer, and data scientist in different companies for several years.