A Framework for Enhancing the Modeling and Comprehension of Declarative Process Models

Abbad Andaloussi, Amine

Publication date:
2021

Document Version
Publisher's PDF, also known as Version of record

Citation (APA):
Abbad Andaloussi, A. (2021). *A Framework for Enhancing the Modeling and Comprehension of Declarative Process Models*. Technical University of Denmark.

A Framework for Enhancing the Modeling and Comprehension of Declarative Process Models

Amine Abbad Andaloussi

Kongens Lyngby 2020

Technical University of Denmark
Department of Applied Mathematics and Computer Science
Richard Petersens Plads, building 324,
2800 Kongens Lyngby, Denmark
Phone +45 4525 3031
compute@compute.dtu.dk
www.compute.dtu.dk

Summary

All organizations around the World strive to deliver high-quality products and services. Business processes are the key instrument to achieve this goal. Through them, business activities can be explicitly defined and performed in a coordinated manner, allowing to create value for all process stakeholders.

Process models are used to capture, conceptualize and represent business processes. Typically depicted in graphical formats, process models allow to foster the understanding of business processes and support more effective communication between the different process stakeholders. However, the inherent complexity of some processes and the complicatedness of the process modeling languages used to represent them visually can interfere with this goal.

Research about the understandability of process models has addressed different aspects, notably the characteristics of different process modeling languages and their ability to express comprehensible process models. In that regard, two key language paradigms were be discerned: imperative languages depicting all the possible execution paths explicitly in the model, and declarative language abstracting individual execution paths and rather specifying the constraints guiding the overall interplay between the process activities. While imperative languages provide adequate representations for pre-specified and repetitive processes (e.g., security checks in airports), declarative languages are rather dedicated to dynamic and flexible processes meant to fit and adapt in different circumstances (e.g., law-based processes). Given the flexible nature of many business processes in our World, declarative languages are good candidates for capturing and represent them in a concise manner.

The flexibility granted by declarative languages requires process stakeholders to adopt a constraint-based approach to describe and interpret process models. From a cognitive perspective, this approach can challenge humans' understanding of process models. To overcome this limitation, a set of tools and approaches have been proposed in the literature. This thesis sheds light on these mechanisms to scrutinize their support and improve their effectiveness during the modeling and comprehension of declarative process models.

The carried-out studies focus on the Dynamic Condition Response (DCR) language, being one of the most adopted declarative languages in industry, that is also supported by a wide range of tools, embedded in an online modeling platform, available for both academic and commercial uses. The DCR platform comprises a graphical editor producing declarative process models, a textual annotator mapping process specifications with the model elements and a simulation tool illustrating the execution paths allowed in the model.

The research conducted as part of this thesis can be organized into three key contributions. In the first contribution, the combination of the aforementioned tools is conceived into a hybrid process artifact, which is itself, is part of a larger family of hybrid business process representations (HBPRs), that have been proposed in the literature to support the modeling and comprehension of process models. The conceptual research is followed by a systematic literature review where similar HBPRs are identified, examined and compared on different levels. In the second contribution, hybrid process artifacts are investigated in terms of their support for process modeling and model comprehension tasks. As for the third contribution, the focus is shifted to the declarative process model within the hybrid process artifact. Herein, a set of modeling practices and complexity metrics are proposed to support declarative process modeling and provide quantifiable means to assess the understandability of declarative models.

The outcome of this thesis contributes to the development of new approaches and tools, providing additional support for the modeling and comprehension of declarative process models and thus promoting their use in practice.

Preface

This thesis summarizes the scientific work conducted from February 15, 2018, to December 15, 2020, to fulfill the requirements for acquiring a Ph.D. in Computer Science from the Technical University of Denmark.

The Ph.D. studies were conducted within the Software and Process Engineering Section at DTU Compute, under the supervision of Assoc. Prof. `Andrea Burattin` (Supervisor), Prof. `Barbara Weber` (Co-supervisor) and Assoc. Prof. `Tijs Slaats` (Co-supervisor). The Ph.D. Project was partially funded by Innovation Fund Denmark, Project EcoKnow (7050-00034A).

The thesis document is written in the format of a *collection of articles*. The included articles (5) are the following:

Articles published in journal venues

- **On the declarative paradigm in hybrid business process representations: A conceptual framework and a systematic literature study.** (Article 1)
  Andaloussi, A. A., Burattin, A., Slaats, T., Kindler, E., & Weber, B.
  Information Systems [ABS^{+}*20], 2020.*
  Summarized in: Part I, Chapter 3
  Article appended in: Part II, Chapter 7

- **Exploring how users engage with hybrid process artifacts based on declarative process models: a behavioral analysis based on eye-tracking and think-aloud.** (Article 3)

Andaloussi, A. A., Zerbato, F., Burattin, A., Slaats, T., Hildebrandt, T. T., & Weber
Software and Systems Modeling [AZB+20], 2020.
Summarized in: Part I, Chapter 4
Article appended in: Part II, Chapter 9

Articles published in conference venues

- **Understanding quality in declarative process modeling through the mental models of experts.** (Article 4)
  Andaloussi, A. A., Davis, C. J., Burattin, A., López, H. A., Slaats, T., & Weber, B.
  International Conference on Business Process Management [ADB+20], 2020.
  Summarized in: Part I, Chapter 5
  Article appended in: Part II, Chapter 10
- **Exploring the modeling of declarative processes using a hybrid approach.** (Article 2)
  Andaloussi, A. A., Buch-Lorentsen, J., López, H. A., Slaats, T., & Weber, B.
  International Conference on Conceptual Modeling [ABLL+19], 2019.
  Summarized in: Part I, Chapter 4
  Article appended in: Part II, Chapter 8

Article submitted as part of the thesis

- **Assessing the complexity of declarative process models using model-based metrics.** (Article 5)
  Andaloussi, A. A., Burattin, A., Slaats, T., Kindler, E., & Weber, B.
  (Reported in the thesis)
  Summarized in: Part I, Chapter 5
  Article appended in: Part II, Chapter 11

The Ph.D. Student (`Amine Abbad Andaloussi`) was the leading author in all the included articles.

Besides the articles included in this document, the Ph.D. Student has led or contributed to a series of other articles (7) published as preliminary work, research related to the Ecoknow project, or general research about the understandability of process models (cf. Section 1.7.2).

This document is organized into two parts: Part I provides a summary of the contributions made during the Ph.D., while Part II includes the pre-print copies of the articles summarized in this document.

Lyngby, 15-December-2020

Amine Abbad-Andaloussi

Amine Abbad Andaloussi

Acknowledgments

I owe a debt of gratitude to all the people who helped me during this fascinating journey. In all its ups and downs, the Ph.D. journey was undoubtedly the most enjoyable experience I ever had. I want to start with Barbara Weber; I think that words cannot express how grateful I am to her. No matter how busy she was, she always could find time to provide guidance and feedback, which were the key ingredients to make this journey as rewarding it is. I want to thank Andrea Burattin for the countless discussions, suggestions, and feedback. I am very grateful for his invaluable support and seemingly endless help, without which this work would never have been possible. I must also express a great acknowledgment to Tijs Slaats for his insightful feedback and continuous willingness to share his experience and answer my questions at any time.

I also thank Ekkart Kindler for his particular help and support on both professional and personal levels. His comments were key to most of the research contributions I made in this thesis, and his advice and insights helped a lot to push my limits and deliver my best.

I want to thank my colleagues at the Software and Process Engineering Section. I have enjoyed every discussion we had and every event we organized. Thanks to them, our work environment was enjoyable and very special, which has contributed a lot to my happiness and enhanced my productivity.

I had the chance to collaborate with many passionate researchers, who I want to thank a lot for all their efforts and impactful contributions. Particular thanks go to the previously mentioned people, Christopher Davis, Thomas Hildebrandt, Hugo López, Pnina Soffer, Francesca Zerbato and all the researchers with whom

I have worked in the past.

Special thanks also go to Manfred Reichert and Hajo Reijers for accepting to join the Ph.D. committee and evaluate my work.

Last but certainly not least, my parents and my sweet sister were the closest people to me during this period. Thank you for your care and endless love and for your greatest gift that is to convenience me that with hard work and patience, I can attain all my aims. I would never have come this far without your emotional support and encouragement.

Contents

Part I

Thesis Summary

CHAPTER 1

Introduction

This chapter provides an introduction for the thesis. Sections 1.1 and 1.2 define the general scope of the thesis. Section 1.3 introduces the research problems. Section 1.4 presents the setting used for the empirical research. Section 1.5 outlines the thesis contributions. Section 1.6 introduces the used research method. Section 1.7 gives an overview of the document content and structure.

1.1 Business Processes Management

Business processes are strategic assets for many industries. By definition, a business process refers to *"a set of activities that are performed in coordination in an organizational and technical environment. These activities jointly realize a business goal"* [Wes19]. In the past decades, there has been an increased interest in understanding the way business processes operate in organizations to evaluate their quality and increase their value [DLRMR13]. In this scope, the discipline of "Business Process Management" (BPM) has emerged. Building upon the existing research in Business Administration, Organization Management and Computer Science, BPM aims at providing an overarching framework comprising *"concepts, methods and techniques to support the design, administration, configuration, enactment and analysis of business processes"* [Wes19]. Bridging the gap between different disciplines, BPM research and tools have quickly become indispensable in many industries and caught the interest of different process stakeholders, including business (i.e., domain) experts and IT specialists [DLRMR13, Wes19, Men07, RRIG09, HSFM06, AG11].

BPM addresses different phases of the business process life-cycle including the process *design and analysis*, *configuration*, *enactment* and *evaluation* [Wes19] (cf. Figure 1.1). In the *design and analysis* phases, the organization's business processes are identified and *modeled* as *process models.* Generally, *"a process model consists of a set of activity models and of execution constraints among them"* [Wes19]. Usually, a process model is represented using one or more process modeling languages, typically providing a graphical representation allowing to visualize the relationships between the different process activities. Process models enable the communication between different stakeholders and support better analysis and improvement of the organization's processes through a set of model-checking, simulation and validation techniques [Wes19]. Furthermore, process models provide a blue-print for process execution.

During *the configuration phase*, the process blue-print is implemented and configured to operate in a Process Aware Information System (PAIS), i.e., a special type of Information Systems powered by one or more process models working in coordination to achieve the intended goals [Wes19]. The *enactment phase* follows the configuration phase. Therein, the configured models are instantiated and executed with the support of the PAIS. At run-time, automated activities can be directly executed by the PAIS, while activities requiring human operators to interact with their physical environment can be traced at coarse-grained level (e.g., by marking their beginning and end) or at fine-grained level (e.g., by using sensors providing real-time information about their progress). As a result, the flow and the progress of these activities can be monitored at run-time, giving, in turn, valuable information to all the stakeholders involved in the process [Wes19].

In addition, process enactment generates a large amount of data which can be analyzed during the *Evaluation phase.* In this phase, the process data is organized into event-logs and analyzed using different process mining techniques [VDA16, Wes19]. These techniques allow to discover the actual control-flow of the process (i.e., the logical order at which the process activities occur), check its conformance with the reference process model (configured in the PAIS) and run a set of predictive analyses allowing to identify patterns in the data and predict the outcome of the future process executions [VDA16].

As shown in Figure 1.1, the research conducted in this thesis fits within the design and analysis phase of the BPM life-cycle. In particular, the focus is on the modeling of business processes. Given its human-intensive nature, *business process modeling* (or shortly *process modeling*) can be seen as one of the most challenging and least controlled activities of the BPM life-cycle [Pin14]. Being fundamental for both the management and enactment of business processes, process models (i.e., the product of process modeling) are required to be intuitive and understandable to all process stakeholders [DVDA04, Rit07]. However, in practice, process models have been associated with several issues, reducing their quality and hindering their understandability [Fig17, WRMR11, Men09]. To overcome these challenges, it is necessary to investigate the modeling and comprehension of process models in order to develop robust approaches and tools supporting process stakeholders and ensuring an enhanced quality of process models.

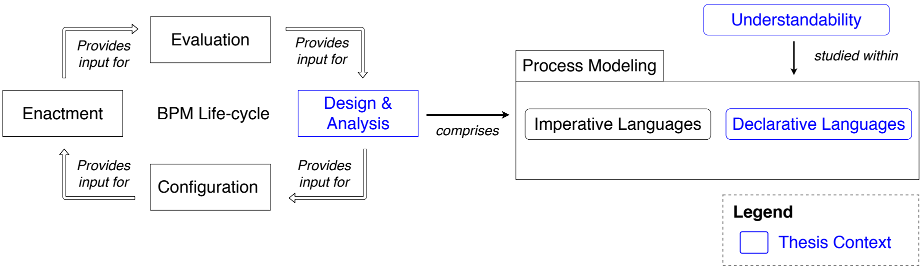


Figure 1.1: Research context of the thesis

1.2 Process Modeling

Business Processes are present in almost all organizations, whether it is a governmental agency, a non-profit organization, an enterprise, their work is always guided by a set of rules, regulations and procedures which define the series of actions that should be taken to achieve particular goals. Capturing, abstracting, conceptualizing and representing business processes as process models is what defines *process modeling*, which is also a key requirement for engaging with any phase of the BPM life-cycle [Men07, DLRMR13].

Business processes are modeled using *process modeling languages*. In the literature, several languages have been proposed (e.g.. Business Process Modeling Notation BPMN [OMG06], Event-driven Process Chains EPC [KNS92], Petri nets [Pet62], Declare [PSVdA07], Dynamic Condition Response DCR [HM11]). These languages have different strengths and weaknesses and the choice of which language delivers understandable process models depends on the nature of the business process being modeled [FLM+09]. Process modeling languages are organized into the *imperative–declarative* paradigm spectrum [FLM+09]. The languages in this spectrum take different approaches to describe the control-flow of business processes. Imperative languages (e.g., BPMN [OMG06]) describe explicitly all the execution paths supported by the process. Reading imperative models, in turn, requires following a sequential order that is typically in line with the temporal and logical order of activities in the process. Such a representation can be easily perceived and understood by readers, especially if the modeled process belongs to the prespecified and repetitive class of business processes. These processes are generally well structured and have a process logic that is rigid and can be concisely prespecified in the model at design time (e.g. check-in procedure for passengers in airports) [RW12]. However, in practice, many business processes are rather dynamic and knowledge intensive, meaning that it is up to the process actors to decide – based on their knowledge – on which actions to take and what order to follow when executing the process. Such processes are common in several areas like insurance, banking, health-care, law and public administration. The number of possible executions paths in these processes is usually very high. Using imperative languages to represent these processes results in "spaghetti"-like models where the process control-flow is convoluted and hard to be perceived and understood by humans [FLM^{+}09]. Alternatively, declarative languages (e.g., Declare [PSVdA07], DCR [HM11]) promise a more concise representation of flexible processes. Following a constraint-based approach, declarative languages abstract from describing all the possible execution paths and rather use constraints to specify the forbidden behavior of the process, leaving all behavior not violating the specified constraints to be allowed by the process model. From this perspective, declarative languages produce concise process models. However, the use of constraints requires a different approach to read and comprehend declarative process models. Unlike imperative process models, their declarative counterparts require the reader to understand the interplay between different constraints and keep track of their implication on the execution of each process activity [FLM^{+}09]. This requirement gets more challenging when considering indirect constraints between activities (i.e., *hidden dependencies*), which in turn require a more holistic approach to understand the model. In other words, the user needs to retain a large amount of information in his memory before he is able to make sense of the process control-flow [Zug13]. Knowing that the human working memory has a limited capacity [Mil56], the user might get quickly overloaded and thus become unable to perceive the interplay of constraints in the model.

All in all, the flexible nature of many business processes in the real-world requires languages that can conceptualize and represent them in a concise manner. Declarative languages are good candidates to meet this requirement [RW12]. However, their constraint-based approach can be seen as a barrier, challenging their adoption in practice. This, in turn, raises the need for investigating novel methods to improve the modeling and comprehension of declarative process models. As shown in Figure 1.1, the context of this thesis relates particularly to the understandability of declarative languages.

1.3 Research Problems

The use of declarative process modeling languages is associated with a series of challenges, which can be attributed to (a) the *modeling* of business processes using declarative languages and (b) the *comprehension* of declarative process models. Section 1.3.1 introduces the modeling challenges, while Section 1.3.2 presents the comprehension challenges. An overview of the challenges introduced in this section is shown in Figure 1.2 (a).

1.3.1 Modeling Business Processes Using Declarative Languages

The modeling of business processes is challenged by the quality of the existing *tool-support* and *modeling practices*. These two aspects have been identified as being pertinent factors affecting the process and the product of process modeling [Pin14, SKW11, PZW$^+$11, Zug13, PPZW12, DV11].

The constraint-based approach of declarative languages requires a different perspective to conceptualize and represent the control-flow of business processes. Modeling following this approach does not only require a graphical modeling interface (or shortly, graphical editor) but also other instruments to support declarative process modeling. Indeed, as declarative languages do not provide the means to depict the process control-flow in a sequential manner [FLM$^+$09], process modelers have to mentally simulate the interplay imposed by the model constraints to ensure that the business process behavior is correctly captured [Zug13]. In addition, the mapping between the process specifications and the language constructs can pose significant challenges during process modeling [SKW11]. This is particularly due to the informal nature of process specifications, which are usually represented in natural languages leaving room for different interpretations [LMMS19]. When modeling using a declarative language, for instance, small language variations in the process specifications (e.g., the use of different modal verbs) might be mapped to different constraints [LDSH20]. As a result, the produced process models may not only convey different meanings for the same business process, but would also lack the mechanisms allowing to track and document modelers' own interpretation of the process constraints and the subsequent design decisions made during process modeling. This, in turn, affects the maintainability and understandability of the produced models and challenge the ability to effectively check their compliance with the process specifications. To overcome these challenges, a few authors in the literature have suggested test-driven modeling approaches combining formal process models and test-cases (e.g., [ZPW11]) and literate process modeling approaches where textual process descriptions are interweaved with formal process models (e.g., [LDHM18, PPZW12]). However, there is still a need for more research about how these approaches support the modeling of business processes.

Modeling practices also occupy a central position in the modeling of business processes [Kro16]. These practices are the product of the guidelines proposed in the literature [MRvdA10, CFF+18]. While a rich body of research investigates the quality of imperative process models, only a few studies address the quality of the declarative ones [HSLW18]. Subsequently, the applicability of existing imperative process modeling guidelines to declarative models remains unclear in the literature. There are several reasons for questioning the applicability of these guidelines for declarative models. On the one hand, imperative and declarative languages use different sets of vocabulary. Moreover, imperative process modeling guidelines provide recommendations which might inhibit the quality of declarative process models. For instance, some guidelines [CFF+18] advise sequencing the process control-flow and minimizing concurrency. In turn, applying these guidelines when modeling declaratively would limit the flexibility of declarative models and over-constrain their behavior. On the other hand, guidelines addressing the aesthetic and the secondary notation of models [CFF+18, Moo09] could be potentially shared between imperative and declarative languages. In order to support declarative process modeling, it is important to recognize the aspects defining the quality of declarative process models. These aspects will guide modelers and provide them with clear modeling practices delineating the proper approach to capture the process control-flow and represent it in a readable manner.

1.3.2 Comprehending Declarative Process Models

The comprehension of declarative process models is another important aspect impeding their adoption [Zug13]. In that regards, the literature lacks comprehensive insights about the *usage* of declarative process models and well-defined complexity *metrics* allowing to assess their understandability.

Investigating the way people seek information from declarative process models is fundamental for tailoring existing process representations and tool-support to the individual need of people. In addition, it enables identifying the benefits and challenges perceived by people when engaging with declarative process models. Recognizing these characteristics, in turn, is deemed necessary for evaluating the effectiveness of existing process modeling guidelines and for providing new insights about the pitfalls that must be addressed in the future.

Besides seeking users' feedback based on their experience while engaging with declarative process models, it is also important to develop objective measures to compare declarative models and assess their complexity. The literature attaches great importance to the development of complexity metrics and highlights their ability to accurately estimate the quality and understandability of imperative process models [PC17, Men07]. When it comes to their declarative counterpart, the literature clearly lacks metrics for assessing the complexity of declarative process models. Defining these metrics is a major step towards an effective evaluation of the understandability of declarative process models and a clear distinction between good and bad process designs, which in turn is crucial for the improvement of existing process models.

1.4 Setting for Empirical Research

The research conducted in this thesis comes as part of the Ecoknow project[1], providing effective, co-created, compliant adaptive PAIS for knowledge workers [HAC$^+$20]. Among the objectives set by the project is to empower municipal employees (with a moderate background in conceptual modeling) to design and maintain models of the law on their own. Such support is deemed important considering the rapid pace at which laws and regulations change and evolve over time and the cost of outsourcing system updates required after each change in the law [HAC$^+$20]. To this end, it is important to provide a language, a modeling approach and a tool-support that can be easily manipulated by novices.

DCR Solutions[2] is a key industrial partner and technology provider in this project. The DCR portal[3] provides a platform for designing and enacting declarative process models expressed in the *DCR language*. This language has many synergies with other declarative languages such as the Case Management Model and Notation (CMMN) [Obj16], Declare [PSVdA07] and its predecessor DecSerFlow [vdAP06]. Nevertheless, the DCR language is rather unique in being fully supported by commercial PAIS systems and deployed by one of the largest technology providers in Denmark, providing solutions to a wide array of Danish central government intuitions[4] [HAC$^+$20].

---

[1] See `www.ecoknow.org`

[2] See `https://dcrsolutions.net`

[3] See `http://dcrgraphs.net`

[4] See `https://www.kmd.dk/indsigter/fleksibilitet-og-dynamisk-sagsbehandling-i-staten`

The DCR Portal offers a suite of tools, including a graphical editor, a textual annotator and a simulation tool [MSS16, LDHM18]. These tools are meant to support both the modeling and comprehension of declarative process models. During modeling tasks, process modelers can use the textual annotator to annotate activities, roles and constraints in the process description, which in are in turn rendered in the graphical editor. Additionally, modelers can intertwine between the graphical editor and the guided simulation to refine and validate the behavior of the process model. As for tasks involving model comprehension, users can refer the process model (shown in the graphical editor) to check the constraints governing the interplay between the process activities, use textual annotations to extract contextual information about activities and constraints, and rely on the guided simulations (offered by the simulation tool) to evaluate whether a certain execution path is supported by the model.

While the platform provides the infrastructure and the instruments for enhanced process modeling and comprehension, there is still a pertinent need for empirical research to (a) evaluate the quality of the proposed instruments, (b) investigate the practices supporting an effective modeling of business processes and (c) understand the factors influencing the understandability of declarative models. All this research shall be conducted within a rigorous scientific framework built upon the state of the art literature in process modeling and cognitive psychology.

The DCR Portal, in turn, provides the settings for the empirical research carried-out in this thesis. The subsequent studies focus on three important *process design artifacts* (or shortly, process artifacts) in the DCR Portal. Namely, the graphical editor, the textual annotator and the simulation tool. These three artifacts are conceived within an overarching conceptual framework and analyzed at different levels of granularity. Figure 1.2(b) illustrates the empirical research settings, while the following section (Section 1.5) explains how this setting is operationalized to investigate the research problems presented in Section 1.3.

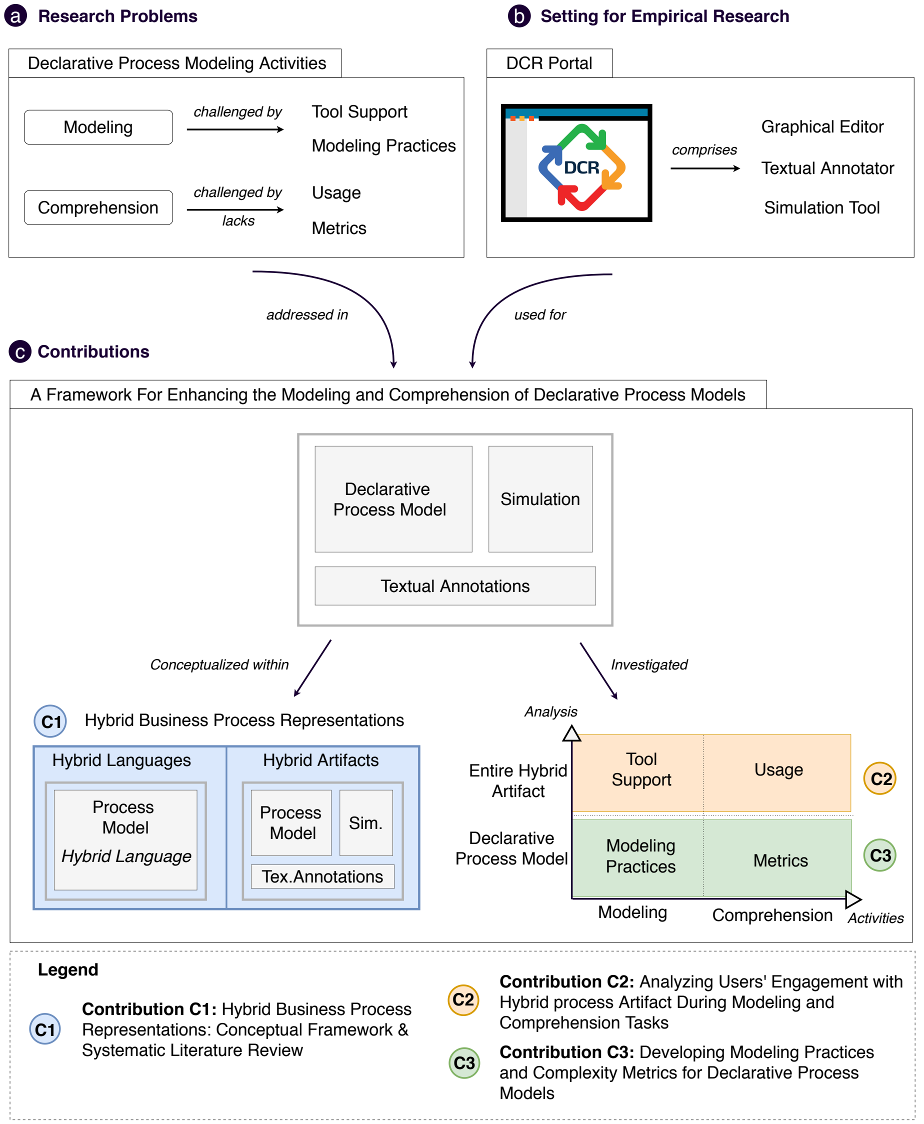


Figure 1.2: Research problems, setting for empirical research and research contributions of the thesis

1.5 Research Contributions

This section presents the research contributions of this thesis, which aims at developing a framework for enhancing the modeling and comprehension of declarative process models. This framework addresses the research problems introduced in Section 1.3 in the context of the empirical research setting presented in Section 1.4. The scientific contributions in this thesis can be organized into two pillars: *conceptualization of hybrid process artifacts* and *investigation of hybrid process artifacts*. The first pillar is addressed in **Contribution C1**, while the second pillar is addressed in **Contributions C2 and C3**. These contributions are illustrated in Figure 1.2 and presented in more detail, respectively, in Sections 1.5.1, 1.5.2 and 1.5.3.

1.5.1 Contribution C1

Contribution C1 consists of a conceptual framework and Systematic Literature Review (SLR) where *hybrid process artifacts* are conceptualized and studied within the wide body of existing process modeling literature.

In the scope of this research, a hybrid process artifact is defined as a process representation combining two or more design artifacts (e.g., process models, textual annotations, guided simulations) overlapping in the description of some aspects of the process [ABS$^+$20, AZB$^+$20]. This definition provides an abstraction for the relationship between the three artifacts covered in the DCR Portal (cf. Figure 1.2(b)) and allows to generalize to other similar "hybrid process artifacts" proposed in the literature. This definitions also comes as part of an overarching conceptual framework where hybrid process artifacts are conceived to be part of *Hybrid Business Process Representations* (HBPRs). This family of process representations comprises *hybrid process artifacts* – but also *hybrid languages*, which refer to languages combined at the level of their syntax, semantics and language paradigm (i.e., imperative, declarative) [ABS$^+$20].

In this contribution, a conceptual framework discerning the relationships between the concepts of a process artifact is proposed. This framework is, in turn, instantiated to deliver a unified terminology where the different types of HBPRs (i.e., hybrid process artifacts and hybrid languages) are conceived. Following that, an SLR is conducted to study exiting hybrid representations, discern their characteristics and organize them into a taxonomy. While the conceptual framework is generic to both imperative and declarative languages and artifacts, the SLR emphasizes the declarative paradigm in HBPRs. Finally, the outcome of the SLR is used to delineate an agenda for future research.

1.5.2 Contribution C2

Contribution C2 consists of studying users' engagement with the *entire DCR hybrid process artifact* composed of the declarative process model (provided by the graphical editor), guided simulations (provided by the simulation tool) and textual annotations (provided by the textual annotator) (cf. Section 1.4). This contribution comprises two studies: the first study addresses the quality of the DCR *tool-support*, provided by the hybrid artifact, during process *modeling* tasks, while the second study addresses the *usage* of the hybrid artifact during *comprehension* tasks.

Process Modeling Using Hybrid Artifact. The first study aims at exploring the *tool-support* granted by hybrid process artifacts during process modeling tasks. As mentioned in Section 1.4, the graphical editor, the textual annotator and the simulation tool are meant to deliver a hybrid process artifact incorporated in a tool-support that is providing additional channels to facilitate the modeling of declarative process models. The study investigates the process in which these artifacts are conjoined and interrelated to turn a process description into a declarative process model. The investigation is supported by the analysis of users' interactions within the DCR Portal and a qualitative coding of verbal utterances obtained from retrospective think-aloud sessions.

In a nutshell, the study reveals the actual use of the hybrid artifact, during process modeling tasks, and explains the rationals behind the observed behaviors from a modeler perspective. In addition, the study weights the benefits and challenges associated with the use of such a hybrid artifact and reports on how it can help to improve the quality of the produced declarative models.

Model Comprehension Using Hybrid Artifact. The second study aims at exploring the *usage* hybrid process artifacts (embedding a declarative process model) during tasks involving model comprehension. The study uses similar artifacts as those deployed in the process modeling study while shifting the focus from process modeling to model comprehension. The study examines how hybrid process artifacts support the extraction of different types of information about business processes and show how users' background and task type influence the way artifacts are perceived and combined during model comprehension. The investigation is supported by a multi-granular analysis of eye-tracking data [HNA$^+$11] and a qualitative coding of verbal utterances obtained from retrospective think-aloud sessions.

All in all, the study delineates an approach to mine and discover the processes guiding users' eye-movements across a visual stimulus (e.g., a hybrid process artifact shown in a computer screen) based on process mining techniques [VDA16].

The approach is, in turn, used to conducted coarse-grained analysis of the common reading patterns exhibited by users with different backgrounds when conducting different comprehension tasks. To gain a deep understanding of the observed reading patterns, the perceived benefits and challenges associated with each artifact are extracted from users' verbal utterances and examined. Finally, a fine-grained analysis of eye-tracking data is conducted to delve further into users' visual behavior and delineate the strategies adopted by users when searching for information across different process artifacts.

1.5.3 Contribution C3

Contribution C3 consists of developing a catalog of modeling practices and a set of complexity metrics for declarative process models. The studies conducted as part of this contribution put the focus on *the declarative process model*, being the core and the target artifact in the hybrid representation. The research conducted within C3 comprises two studies: the first study investigates the *modeling practices* guiding process *modeling* tasks, while the second study develops and evaluates a set of complexity *metrics* allowing to appraise the *comprehension* of declarative process models.

Modeling Practices. The first study within this contribution aims at eliciting a set of modeling practices to guide modelers during process modeling. These practices have emerged from a series of consultations with experts in process modeling. Supported by the Personal Construct Theory (PCT) [Kel55], the criteria used by experts to judge the quality of declarative process models are articulated and scrutinized to develop an overarching understanding of quality in the context of declarative process modeling.

The first part of the study consists of adapting PCT and adjusting the underlying approach for the intended research. The second part, in turn, uses the adjusted approach to develop a catalog of modeling practices intended to support modelers during process modeling and enhance the quality of the produced process models.

Complexity Metrics. The second study within this contribution aims at developing and evaluating a set of complexity metrics providing quantifiable means to assess the understandability of declarative process models. These metrics are derived from those proposed in imperative process modeling [Men07] and investigated from a human-cognitive perspective.

In the first part of the study, the complexity metrics are defined together with a set of hypotheses stipulating the impact of the factors, addressed by these metrics, on users' cognitive load. The second part, in turn, consists of designing and conducting an experiment [WRH$^+$12] where the candidate hypotheses are tested and validated empirically.

1.6 Research Method

The research conducted in this thesis is guided by *qualitative* and *quantitative* research methods. Sections 1.6.1 and 1.6.2 introduce these methods respectively and provide an overview on their use in the context of this thesis.

1.6.1 Qualitative Methods

Qualitative research methods represent a class of systematic approaches guided by inductive processes for analyzing and organizing data into categories and themes which, in turn, serve as a basis for identifying new patterns and provide the grounds for raising new hypotheses and developing new theories [MS93]. Qualitative methods are typically used in exploratory contexts where phenomena are observed, scrutinized and understood in their natural setting. This type of research is usually based on interviews, observations and document reviews [Ast13].

Qualitative research comprises different methodologies, notably, grounded theory. Through this methodology, hypotheses and theories are emerged from the data [CB07, Ast13]. The approach underlying grounded theory involves a meticulous data analysis guided by a set of coding techniques performed in incremental coding cycles that eventually lead to the construction of a network of codes summarizing the pertinent aspects in the data [Sal15].

In the scope of this thesis, grounded theory is used to guide the qualitative analyses conducted in Contributions C2 and the first study in Contribution C3. Overall, the analyses cover both verbal and eye-tracking data and use a series of coding techniques where data is extracted, compared, grouped and overlapped in order to identify the commonalities among the investigated concepts [CB07]. In addition, method triangulation [DN70] is used to compare and validate the findings ensuing from different modalities. This is particularly the case for the study, including eye-tracking (i.e., second study in Contribution C2). Although, the eye-mind hypothesis claims a close relationship between users' visual behavior and the underlying cognitive processes [JC80], users' behavior can be subject to different interpretations and hence can lead to different conclusions [HNA+11]. By triangulating the eye-tracking findings with users' verbal utterances, observations can be interpreted on more solid grounds.

Besides, the SLR study conducted in Contribution C1 can be seen as a study of qualitative nature performed following the SLR methodology of Kitchenham [KC07].

1.6.2 Quantitative Methods

Quantitative research methods refer to a class of systematic approaches mostly guided by deductive processes where the effect of some manipulation or activity on a certain phenomenon is examined in controlled environments [WRH+12]. Quantitative methods are typically used in explanatory studies, where it is important to identify a causal relationship between a group of variables. This type of studies usually requires the design of controlled experiments, where all the factors susceptible to influence the outcome of the study are either fixed or controlled. Such a design supports comparisons and statistical analyses [WRH+12].

Quantitative research comprises a broad range of analysis approaches, notably, hypothesis-testing techniques. Through a variety of statistical tests, these techniques determine whether there is a significant effect between a set of independent and dependent variables [WRH+12].

In the context of this thesis, a hypothesis-testing technique is used in the second study of Contribution C3 to test a series of hypotheses in a controlled experiment. The candidate hypotheses stipulate a causal relationship between a set of factors (captured using the proposed complexity metrics) and a set of estimators of cognitive load (e.g., perceived difficulty, comprehension accuracy, response time).

1.7 Thesis Overview

This thesis consists of a collection of several articles covering the research completed during this Ph.D. Section 1.7.1 presents the structure of this document and delineates the articles within each contribution, while Section 1.7.2 provides a brief summary of other articles published as part of this Ph.D. project or within the wider stream of research about the understandability of process models. An overview of the articles (i.e., articles' titles, authors and venues) included in this thesis and the other articles published during the Ph.D. is shown in Figure 1.3.

1.7.1 Thesis Structure and Articles

The thesis is structured into two parts. Part I contains an introduction (current chapter), a background (cf. Chapter 2), three main chapters summarizing five articles representing the core contributions of the thesis (cf. Chapters 3, 4 and 5) and a conclusion (cf. Chapter 6). Part II includes the pre-print versions of the covered articles.

The contents of the chapters in Part I are described below:

- **Chapter 2** provides a background on the key notions discussed in the thesis. In particular, the chapter covers the existing process modeling paradigms (i.e., imperative, declarative) and the declarative language investigated in this work (i.e., DCR Graphs [HM11]). In addition, the chapter delves into the field of cognitive psychology to highlight the limitations of humans' working memory and discuss some approaches to mitigate these limitations.
- **Chapter 3** summarizes Contribution *C1* (cf. Section 1.5.1). In particular, an overview of the conceptual framework laying the foundations for defining and characterizing HBPRs is provided. Following that, a summary of the results of the SLR exploring the existing literature about HBPRs is presented. The full contribution is reported in *Article 1* (cf. Chapter 7).
- **Chapter 4** summarizes Contribution *C2* which comprises two studies (cf. Section 1.5.2). The first study investigates process modeling with the support of hybrid process artifacts, while the second study examines model comprehension using hybrid process artifacts. The studies are reported in detail in *Article 2* (cf. Chapter 8) and *Article 3* (cf. Chapter 9) respectively.
- **Chapter 5** summarizes contribution C3 (cf. Section 1.5.3) which consists of two studies. The first study investigates declarative process modeling practices, while the second study develops and evaluates a set of complexity metrics for declarative models. The studies are reported in detail in *Article 4* (cf. Chapter 10) and *Article 5* (cf. Chapter 11) respectively.
- **Chapter 6** summarizes the key findings and concludes the thesis.

1.7.2 Other Articles

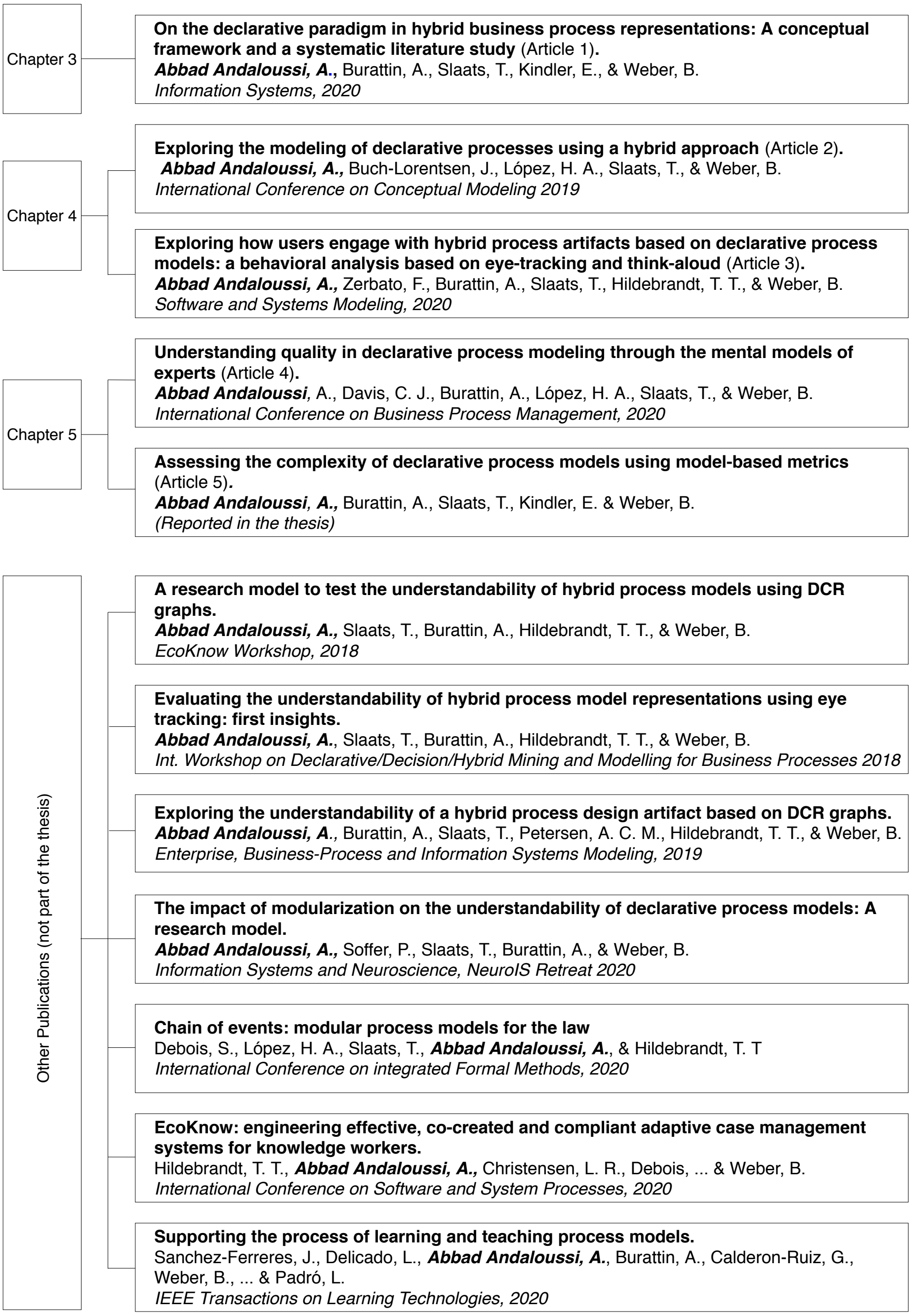
Besides the articles covered in this thesis, many other articles have been published during the Ph.D. studies.

The first publication in the Ph.D. [ASB+18a] comprises a research model delineating the approach meant for investigating the use of the DCR hybrid process artifact during model comprehension tasks. This research model has been refined and adapted to design and conduct the model comprehension study presented in Contribution 2 (cf. Section 1.5.2). In [ASB+18b], first insights resulting from the analysis of users eye-tracking data (when engaging with the DCR hybrid artifact) have been reported, while in [ABS+19], more advanced analyses of both eye-tracking and verbal data have been completed. The results of these three articles have been complemented with a more fine-grained analysis of eye-tracking data and published in [AZB+20] (Article 3), which is included in the thesis document.

Besides this line of publications, a research model has been proposed in [ASS+20] to investigate the impact of modularization on the understandability of declarative process models. Moreover, a qualitative study investigating the use of a novel modularization approach within the DCR Portal has been published as part of the research in [DLS+20]. In addition, a summary of the overall research conducted during the Ph.D. in the context of the Ecoknow project has been reported in [HAC+20].

Finally, an empirical study has been conducted to evaluate a new approach supporting the transition from textual process descriptions to imperative process models in BPMN. The evaluation is published as part of the research in [SFDA+20].

The articles presented in this section are illustrated in the second part of Figure 1.3 (Other Publications), in the same order as they are mentioned in the text.



Chapter 3

On the declarative paradigm in hybrid business process representations: A conceptual framework and a systematic literature study (Article 1).
Abbad Andaloussi, A., Burattin, A., Slaats, T., Kindler, E., & Weber, B.
Information Systems, 2020

Chapter 4

Exploring the modeling of declarative processes using a hybrid approach (Article 2).
Abbad Andaloussi, A., Buch-Lorentsen, J., López, H. A., Slaats, T., & Weber, B.
International Conference on Conceptual Modeling 2019

Exploring how users engage with hybrid process artifacts based on declarative process models: a behavioral analysis based on eye-tracking and think-aloud (Article 3).
Abbad Andaloussi, A., Zerbato, F., Burattin, A., Slaats, T., Hildebrandt, T. T., & Weber, B.
Software and Systems Modeling, 2020

Chapter 5

Understanding quality in declarative process modeling through the mental models of experts (Article 4).
***Abbad Andaloussi**, A.*, Davis, C. J., Burattin, A., López, H. A., Slaats, T., & Weber, B.
International Conference on Business Process Management, 2020

Assessing the complexity of declarative process models using model-based metrics (Article 5).
Abbad Andaloussi, A., Burattin, A., Slaats, T., Kindler, E. & Weber, B.
(Reported in the thesis)

Other Publications (not part of the thesis)

A research model to test the understandability of hybrid process models using DCR graphs.
Abbad Andaloussi, A., Slaats, T., Burattin, A., Hildebrandt, T. T., & Weber, B.
EcoKnow Workshop, 2018

Evaluating the understandability of hybrid process model representations using eye tracking: first insights.
Abbad Andaloussi, A., Slaats, T., Burattin, A., Hildebrandt, T. T., & Weber, B.
Int. Workshop on Declarative/Decision/Hybrid Mining and Modelling for Business Processes 2018

Exploring the understandability of a hybrid process design artifact based on DCR graphs.
Abbad Andaloussi, A., Burattin, A., Slaats, T., Petersen, A. C. M., Hildebrandt, T. T., & Weber, B.
Enterprise, Business-Process and Information Systems Modeling, 2019

The impact of modularization on the understandability of declarative process models: A research model.
Abbad Andaloussi, A., Soffer, P., Slaats, T., Burattin, A., & Weber, B.
Information Systems and Neuroscience, NeuroIS Retreat 2020

Chain of events: modular process models for the law
Debois, S., López, H. A., Slaats, T., ***Abbad Andaloussi, A.***, & Hildebrandt, T. T
International Conference on integrated Formal Methods, 2020

EcoKnow: engineering effective, co-created and compliant adaptive case management systems for knowledge workers.
Hildebrandt, T. T., ***Abbad Andaloussi, A.,*** Christensen, L. R., Debois, ... & Weber, B.
International Conference on Software and System Processes, 2020

Supporting the process of learning and teaching process models.
Sanchez-Ferreres, J., Delicado, L., ***Abbad Andaloussi, A.***, Burattin, A., Calderon-Ruiz, G., Weber, B., ... & Padró, L.
IEEE Transactions on Learning Technologies, 2020

Figure 1.3: Overview of the scientific work completed during the Ph.D.

CHAPTER 2

Background

This chapter provides a background covering the key notions associated with this thesis. Section 2.1 introduces process modeling, while Section 2.2 presents pertinent concepts from cognitive psychology.

2.1 Process Modeling

Process modeling is a human activity that consists of abstracting, conceptualizing and representing business processes as process models [Men07, DLRMR13]. Process models, in turn, represent a set of activity models and execution constraints defining their interplay [Wes19]. As mentioned in Section 1.2, process models can be expressed using languages from the imperative-declarative paradigm spectrum. In this section, more background is provided about the difference between imperative and declarative languages within this paradigm spectrum (cf. Section 2.1.1). Following that, the focus is moved to declarative languages, in particular to the DCR (Dynamic Condition Response) language, which is introduced in more detail (cf. Section 2.1.2).

2.1.1 The Process Modeling Paradigm Spectrum

This section discerns imperative and declarative languages. The explanation is adapted from *Article 1* (cf. Chapter 7, [ABS$^+$20]).

Languages can be compared across different dimensions, including for instance, the formality of their syntax, the formality of their semantics, and their language paradigm. A language paradigm can be viewed as the style in which a language is formulated. Existing languages can be organized in the imperative-declarative paradigm spectrum. This paradigm originates from the computer programming field.

Winograd [Win75] asserts that imperative programming is developed on the principle that *"the knowledge of a subject is intimately bound with the procedures for its use"*. Simply put, imperative programming consists of delineating explicitly the commands mapping an input to a particular output. Conversely, declarative programming, as seen by Lloyd [Llo94], is an approach where one specifies *"what is to be computed, but not necessarily how it is to be computed"*. In other words, declarative programming languages aim at stating the end-goal requirements and letting the system find a way to attain them.

According to Roy and Haridi [VRH$^+$04], the notion of state is central to the distinction between imperative and declarative languages. A state is defined as *"a sequence of values in time that contains the intermediate results of a desired computation"* [VRH$^+$04]. Roy and Haridi assert that states can be implicit or explicit. Therein, an implicit state is seen as a state that resides only in the programmer's mind, meaning that, neither the program nor the computational model are aware of it [VRH$^+$04]. In contrast, an explicit state is perceived as a state which can be explicitly traced in the program and its computational model. Along this line, imperative languages represent states explicitly in the model, whereas their declarative counterpart encode states implicitly in the model.

Roy and Haridi's distinction between imperative and declarative languages is limited to the representational layer of languages. Indeed, at the execution layer, the language is, eventually, interpreted as a series of deterministic procedures. Considering both representational and execution layers of languages, an imperative language can be defined as a language where the states are explicit in the representational and execution layers. In turn, a declarative language can be defined as a language where the states are implicit in the representational layer, while being explicit in the execution layer [ABS$^+$20].

The distinction between implicit and explicit states can be transposed to process modeling languages. Following the definition by Pesic [PSVdA07], within the representation layer, imperative languages (e.g., BPMN [OMG06], Petri-nets [Pet62]) provide process representations where *"all execution alternatives are explicitly specified"*, which in turn, delineate all the underlying states explicitly. Modeling using imperative languages, in turn, requires specifying all these execution paths in the model in an explicit manner. This requirement is only feasible when comprehensive knowledge is available about all the possible (discrete) execution paths that the business process can undergo. In practice, this requirement is not always satisfied. This is particularly the case for flexible processes (e.g., knowledge-intensive processes) in which the process execution depends on contextual information varying from one case to another and usually available only at run-time [SSO01]. Alternatively declarative languages (e.g., Declare [PSVdA07], DCR Graphs [HM11]) provide process representations where *"constraints implicitly specify [the] execution alternatives as all alternatives that satisfy the constraints"* [PSVdA07]. Simply put, in declarative languages, constraints are used to prescribe the interplay between the process activities without being explicit about the sequence of states underlying each execution path. These states are in fact, implicit only in the representational layer of the language, while in the execution layer, the states become explicit as the language predicates are being interpreted. This feature allows declarative languages to capture flexible processes without explicitly specifying all the possible execution paths, which in turn results in more concise process models.

Declarative languages are good candidates for modeling flexible processes. However, their constraint-based approach can challenge their understandability. This limitation is discussed from a cognitive perspective in Section 2.2.1.

2.1.2 Dynamic Condition Response Graphs

This section introduces DCR Graphs based on the background provided in *Article 4* (cf. Chapter 10, [ADB$^+$20]). The notions presented in this section cover those referred to in the different articles included in this thesis.

DCR is a declarative process modeling language [HM11]. A DCR Graph comprises activities and relations. Activities can be assigned one or many roles defining the set of actors allowed to execute them. To maximize flexibility, any activity that is not constrained in the model can be executed at any time and any number of times. Similarly, interrelated activities forming a weakly connected *component* in the graph can be executed without being influenced by other activities outside the weakly connected component (cf. Article 5, Chapter 11). Activities have a marking (state) that is a tuple with three Boolean values: *executed, included* and *pending.* The *executed* marking specifies that the activity has been done at least once in the past. The *included* marking shows that the activity is relevant for the process. The absence of this marking on an activity implies that the activity is *excluded*, meaning that it can neither be executed nor constrain the execution of other activities. Lastly, the *pending* marking indicates that the activity must be eventually executed in the future. This kind of marking is used to set a requirement that must be fulfilled before the process can end.

DCR has six relations[1]. In the following, we use the terms “source activity” and “destination activity” two indicate two activities linked with a DCR relation represented as a directed edge from an activity (at the source) to another activity (at the destination). A *condition* (drawn as an orange Arrow To Dot) constrains the execution of a destination activity by specifying that it cannot be executed before a source activity has been executed at least once in the past. A *milestone* (drawn as a purple Arrow To Diamond) constrains the execution of a destination activity by specifying that while a source activity is pending (i.e., required to be fulfilled), the destination activity cannot be executed. The *inclusion* (drawn as a green Arrow To Plus sign) and *exclusion* (drawn as a red Arrow To Percentage sign) allow a source activity to make a destination activity, respectively, relevant or irrelevant in the process by toggling its *included* marking. Finally, the *response* (drawn as a blue Dot To Arrow) and the *no-response* (drawn as a brown Dot To “x” character) allow a source activity to make a destination activity, respectively, pending or not anymore pending by toggling its *pending* marking. The *inclusion*, *exclusion*, *response* and *no-response* relations exhibit a dynamic behavior in the process model. In that respect, they are not perceived as typical constraints, but rather as relations allowing to capture *effects* that some process activities impose on others. Relations and activities can be combined to represent specific modeling patterns.

Several extensions have been proposed to complement the core DCR language. For instance, hierarchy was introduced through the concept of nesting [HMS11] to enable grouping several activities together (into a *nest* activity) and then append a single relation from or to all of them. Nesting is a shorthand for appending a relation to each individual activity and thus it can be seen as a syntactic sugar within the DCR language. *Multi-instance sub-processes* [DHS18], in contrast, makes significant extensions to the language by supporting the modeling of sub-process templates, which allow instantiating a block of activities (as a sub-process) many times during the process execution. For that purpose, an additional relation called the *spawn* (drawn as a black Arrow To Star) is used to make a source activity instantiate a sub-process (at the destination) in the

---

[1] The DCR language was initially introduced with four basic relations (i.e., condition, response, include and exclude) [HM11] and has been extended with two additional relations (i.e., milestone [HMS11] and no-response), which do not formally add to the expressiveness of the language, but allow to straightforwardly express patterns that are commonly found in practice.

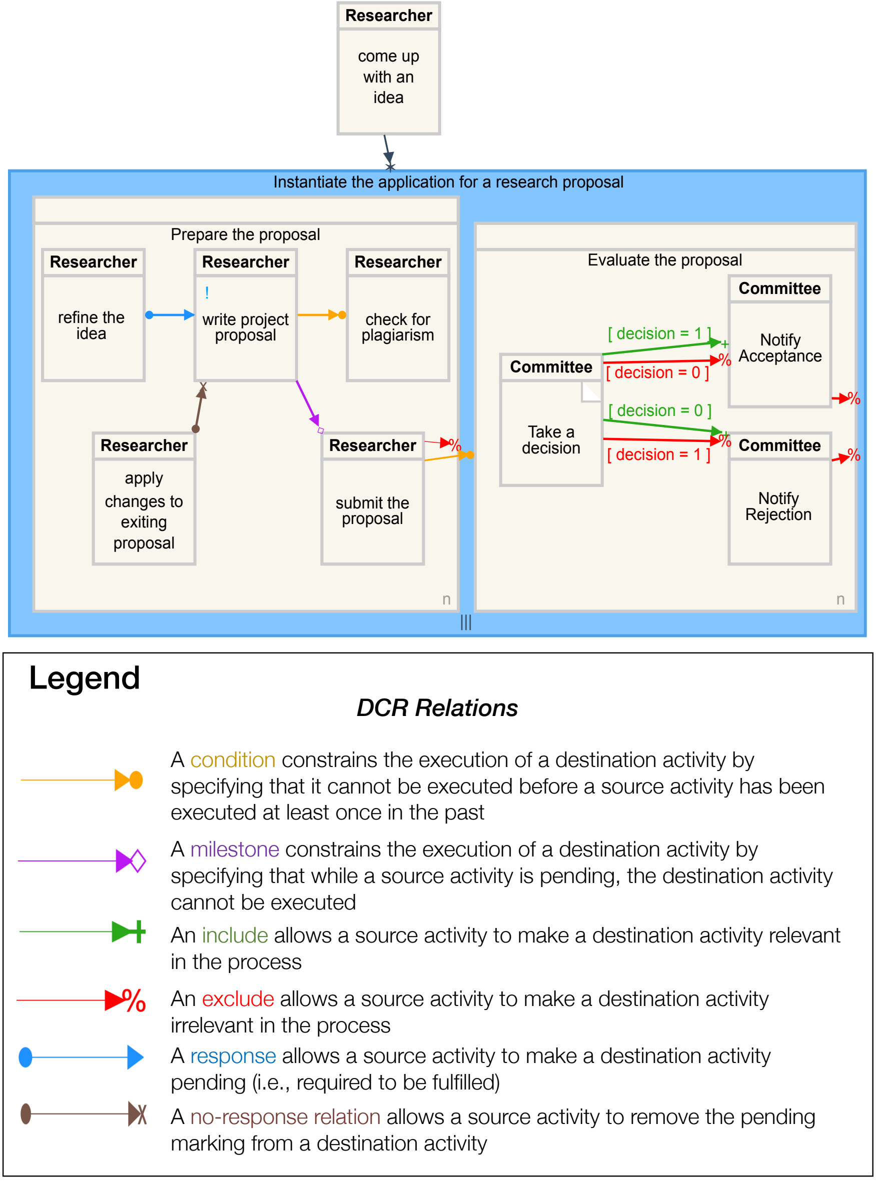
model. Furthermore, DCR allows modeling the effect of contextual *data* on the process control-flow. This support is operated by *data expressions* which can be appended to relations in order to specify the circumstances under which they should be activated [SMHM13]. Data expressions evaluate the values of data variables that are set using a special type of activities called *data activities.* These activities comprise input fields allowing users to assign values to data variables at run-time.

The core DCR language and the introduced extensions are illustrated in the following paragraphs based on the process description shown in Example 2.1 and the respective process model depicted in Figure 2.1.

EXAMPLE 2.1 *The process of writing a project proposal can be described as follows:*

- ***S1.*** *Every time a researcher comes up with an idea, he can instantiate the application for a research proposal.*
- ***S2.*** *Afterwards, the researcher can prepare the proposal. Therein, it is possible for the researcher to write the project proposal, apply changes to an existing proposal or refine his idea at any time.*
- ***S3.*** *Once the research has written the proposal he can check for plagiarism.*
- ***S4.*** *In case the researcher decides to refine his idea, he must write the project proposal again or at least apply some changes to the existing proposal before he can submit it.*
- ***S5.*** *The researcher can submit the project proposal only once.*
- ***S6.*** *After the proposal is submitted, a research committee evaluates the proposal. Thereafter, the research committee takes a decision and notifies the researcher with either an acceptance or a rejection of the proposal.*
- ***S7.*** *After the research committee sends an acceptance or rejection notification, the evaluation of the proposal ends.*

Explanation of Example 2.1. The textual description lists the specifications (Numbered S1–S7) associated with the process of writing a project proposal. The model depicted in Figure 2.1 shows a possible representation of that process. **S1** makes reference to two activities, i.e., *come up with an idea* and *instantiate the application for a research proposal.* The process specification implies that a new process instance should be created every time a researcher comes up with a new idea. Therefore, *instantiate the application for a research proposal* is



DCR Relations

A condition constrains the execution of a destination activity by specifying that it cannot be executed before a source activity has been executed at least once in the past

A milestone constrains the execution of a destination activity by specifying that while a source activity is pending, the destination activity cannot be executed

An include allows a source activity to make a destination activity relevant in the process

An exclude allows a source activity to make a destination activity irrelevant in the process

A response allows a source activity to make a destination activity pending (i.e., required to be fulfilled)

A no-response relation allows a source activity to remove the pending marking from a destination activity

Figure 2.1: The process of writing a project proposal (cf. Example 2.1) modeled using the DCR language. Adapted from [ADB$^+$20], Article (1) and [RW12].

modeled as a sub-process and a *spawn* relation is set between *come up with an idea* and *instantiate the application for a research proposal.*

S2 makes reference to four activities, i.e., *prepare the proposal*, *write project proposal*, *apply changes to existing proposal* and *refine the idea.* The process specification implies that *prepare the proposal* is a nest for the other activities. This is cause *write project proposal*, *apply changes to existing proposal* and *refine the idea* are parts of the preparation of the proposal.

S3 makes reference to two activities i.e., *write project proposal* and *check for plagiarism.* The process specification implies a *condition* relation from *write project proposal* to *check for plagiarism.* This is because *check for plagiarism* cannot be done without having *write project proposal* performed at least once in the past.

S4 makes reference to four activities i.e., *refine the idea* and *write project proposal*, *apply changes to existing proposal* and *submit the proposal.* The process specification implies a *response* relation from *refine the idea* to *write project proposal.* This is because *refine the idea* requires the researcher to *write project proposal.* This requirement can be omitted in case he decides to *apply changes to existing proposal*, which is modeled using a *no-response* relation from *apply changes to existing proposal* to *write project proposal.* The process specification implies also a *milestone* relation between *write project proposal* and *submit the proposal.* This is because, while *write project proposal* is required, the researcher cannot submit the proposal. In this process specification (S4), one can notice two hidden dependencies (i.e., implicit constraints). The first hidden dependency comes between *refine the idea* and *submit the proposal.* Indeed, *submit the proposal* is not only affected by the activity that is directly connected to it (i.e., *write project proposal*), but it is also affected by *refine the idea*, such that if the researcher decides to *refine the idea*, then *write project proposal* becomes pending and subsequently *submit the proposal* gets blocked. A similar hidden dependency can be observed between *apply changes to existing proposal* and *submit the proposal.* This process specification also, denotes the dynamic behavior exhibited by some DCR relations. In this example, for instance, *submit the proposal* can switch between being permitted and blocked several times during the process execution depending on the effect of the *response* and *no-response* relations on *write project proposal.*

S5 makes reference to one activity i.e., *submit the proposal* while the process specification implies an *exclude* relation from *submit the proposal* to the nest activity *prepare the proposal.* Indeed, since the researcher can submit the proposal only once, all the activities inside the nest *prepare the proposal* become irrelevant when the proposal is submitted. The use of an *exclude* relation from an activity to its parent nest activity allows excluding the whole nest after the source activity is performed. This modeling pattern is equivalent to adding a set of *exclude* relations from one activity to (a) itself and to (b) all the other activities in the nest.

S6 makes reference to five activities i.e., *submit the proposal*, *evaluate the proposal*, *take a decision*, *notify acceptance* and *notify rejection.* The process specification implies a *condition* relation from *submit the proposal* to the nest activity *evaluate the proposal*, which comprises the activities associated with the proposal evaluation. This is because the proposal cannot be evaluated before being submitted. Regarding the specification about the committee decision, the outcome of the data activity *take a decision* can either lead to *notify acceptance* or *notify rejection.* This specification is modeled using *data expressions* and a set of *include* and *exclude* relations. Each of these relations takes effect only when the associated expression evaluates to true. For instance, in case the committee decision in positive (i.e., $decision = 1$), the data expressions associated with the *include* relation from *take a decision* to *notify acceptance* and the *exclude* relation from *take a decision* to *notify rejection* will evaluate to true. This, in turn, will include *notify acceptance* and exclude *notify rejection.* Conversely, in case the decision is negative (i.e., $decision = 0$), the data expressions associated with the *exclude* relation from *take a decision* to *notify acceptance* and the *include* relation from *take a decision* to *notify rejection* will evaluate to true. This, in turn, will exclude *notify acceptance* and include *notify rejection.* Note that in the model shown in Figure 2.1 the committee can change their decision (i.e., by performing *take a decision* several times) as long as the notification is not yet sent.

Finally, **S7** makes reference to three activities i.e., *notify acceptance*, *notify rejection* and *evaluate the proposal.* The process specification implies an *exclude* relation from *notify acceptance* to the nest activity *evaluate the proposal*, and another *exclude* relation from *notify rejection* to the nest *evaluate the proposal.* This is because after either activities the evaluation of the proposal ends.

2.2 Cognitive Psychology Within Process Modeling

Cognitive psychology explores the way people perceive, acquire, recall and reason about information [SS16]. The study of humans' memory is among the pertinent research topics addressed in cognitive psychology [SS16]. Over the last century, extensive research has been conducted to investigate the functionalities and characteristics of different types of memory [WN65, AS68, BH74]. Notably, two types of memory have been discerned: the *working memory* responsible for holding and processing information for a short period of time and the *long-term memory* capable of storing information for long or indefinite time spans [SS16, Zhe17, BH74]. Attention has been drawn to the *working memory* as this type of memory is crucial for thinking, learning and comprehension. However, it is constrained by its limited capacity to hold information. The following sections illustrate this limitation (cf. Section 2.2.1) and discuss different approach to mitigate it (cf. Section 2.2.2).

2.2.1 Limitation of Humans' Working Memory

Working memory is crucial for most of our daily activities. It provides a working place for temporary maintaining, integrating and processing information [Bad12, Cow16, Obe09, UE07]. The "magical number of seven" is largely cited as being the number of items which one can retain at a time [Mil56], although other studies claim that our memory cannot retain more than 4 items at a time [Cow01]. Nevertheless, all these studies agree that our working has a limited capacity for maintaining and processing information and therefore, it required to use it efficiently.

During process modeling, working memory is used to store task-relevant information, including those presented to the modeler (e.g., a fragment of a textual process description) and those buffered from long-term memory (e.g., the use of modeling pattern in DCR Graphs). This information is processed and integrated together in the working memory and translated to a set of logical decisions and actions (e.g., establishing a mapping between the textual fragment and the adequate modeling pattern) [Pin14]. Likewise, when reading a process model, working memory is used to recognize the visual configuration of the model symbols (e.g., two boxes, one containing the other and a red arrow from the inner box to the outer box), which are in turn integrated with the knowledge buffered from the long-term memory (e.g., a box denotes an activity, a box containing another box denotes a nest activity and a red arrow denotes an exclude relation) to derive a proper understanding of the presented information (e.g., two boxes, one containing the other and a red arrow from the inner box to the outer box is used to exclude all the activities within a nest activity at once).

While modeling and comprehending simple patterns seems feasible, other tasks may exceed the capacity of one's working memory. This is particularly the case when dealing with declarative process models where constraints form hidden dependencies between activities. Therein, to evaluate whether an activity is permitted, required or blocked, one must not only retain the effect of directly linked activities, but also the influence of the indirectly linked ones. For instance, in the model shown in Figure 2.1, the activity *submit the proposal* is not only affected by *write project proposal* (i.e., directly related) but also by *refine the idea* and *apply changes to existing proposal* (i.e., indirectly related). Another case is when trying to evaluate whether an execution path is allowed in a declarative model. During this task, one must evaluate the interplay between the process activities (similar to the previous case) but also bear in mind that the (direct or indirect) effects of some activities on others might change several times over the course of execution. To this end, one must retain and mentally update the marking of the model activities after each step of the process execution. For instance, in the model shown in Figure 2.1, one could start by performing *write project proposal* and thus *submit the proposal* becomes permitted. Then, before doing it, one might perform *check for plagiarism* and then *refine the idea*. At this point *submit the proposal* becomes not anymore possible. Such a dynamic behavior adds up to the challenges of dealing with declarative process models, particularly those expressed in the DCR language.

These cases suggest that some tasks require more memory than others, but how does working memory influence model quality (during modeling tasks) or users' comprehension abilities (during comprehension tasks)? This question is answered by the *Cognitive Load Theory* (CLT) [Swe11]. CLT is based on the notion of *cognitive load*, which aims at quantifying the extent of demands imposed by certain tasks on humans' mental resources [CZW+16]. More formally, it is defined as *"a multi-dimensional construct representing the load imposed on the working memory during* [the] *performance of a cognitive task"* [CZW+16, PTTVG03, PVM94]. Cognitive load has, in turn, been associated with humans' perceived difficulty and performance, positing that as cognitive load increases, people feel more challenged, become slower and more prone to commit mistakes [CZW+16]. These effects can also occur when dealing with complex process models. To address this challenge, it is important to maintain ones' cognitive load at a moderate level. The following section (Section 2.2.2) discusses different approaches to attain this objective.

2.2.2 Overcoming the Limitation of Humans' Working Memory

Overcoming the limitation of working memory during process modeling and comprehension tasks requires approaches to reduce cognitive load. However, as its definition states, cognitive load is a multi-dimensional construct (cf. Section 2.2.1), meaning that it is defined by several components. To develop robust approaches, it is, therefore, important to identify which component of cognitive load to address.

CLT provides a model of cognitive load defined by three-component factors.

Namely *intrinsic load*, *extraneous load* and *germane load* [SVMP98,PTTVG03]. Intrinsic load is associated with the inherent complexity of the information being processed. When performing a modeling or a comprehension task, intrinsic load would emerge from the inherent complexity of the business process [CZW$^+$16]. Extraneous load is associated with the representation of the task [CZW$^+$16]. In the context of modeling and comprehension tasks, extraneous would emerge, for instance, from the complexity of the tool supporting the task or the way the model is represented to the user. Lastly, germane load is associated with humans' ability to build the appropriate mental schemes to organize information efficiently [CZW$^+$16]. Therein, users can experience different levels of germane load depending on their familiarity with the process modeling tools and practices.

Reducing cognitive load requires addressing its components. In process modeling, intrinsic load varies depending on the complexity of the process specifications. Extraneous load, in contrast, can be reduced by improving the tool-support and the representation of the process model. Likewise, germane load can be reduced by teaching users how to develop the appropriate mental schemes to organize information efficiently when creating and reading process models (e.g., with the support of a catalog of modeling practices). In cognitive psychology, different theories and approaches could contribute for attaining these aims.

Dual-coding. The dual-coding theory postulates that verbal information (e.g., textual annotations) and visual information (e.g., fragments of process models) are processed along separate channels (i.e., the phonological loop and the visuo-spatial sketchpad), which do not complete in the working memory [Pai91]. Hence, information can be effectively conveyed when encoded, both textually and graphically [Moo09]. In this context, hybrid process artifacts can improve process representations by interweaving both textual and graphical artifacts.

Cognitive Fit. The cognitive fit theory posits that a fit between the task requirements and the visual representation of the information is associated with reduced cognitive load and better performance [Ves91]. Hybrid process artifacts can provide cognitive fit for different task types. For example, when evaluating whether an execution path is supported in the model, a guided simulation can create a better cognitive fit compared to a declarative process model. Moreover, when checking whether all process specifications are represented in the model, an explicit visual mapping between the model elements and the textual specifications (obtained with textual annotations) can create a better cognitive fit compared to an implicit mapping (where the process description is not annotated).

Cognitive Offloading. The concept of cognitive offloading [RG16] is based on the fundamental distinction between internal and external representations of information [BGP06, SR96, SR05, Tve02, TS09]. While, internal representations are created in the memory, external representations are rather concretized using physical symbols (e.g, a text, a sketch or a graphical model). Cognitive offloading suggests that externalizing information helps us to overcome the capacity limitation of the working memory and to reduce the computational effort required to integrate new information in the mind [RG16]. When reading this document, for instance, instead of retaining all the pertinent ideas, you can alternatively highlight them in the text or take short notes of them. This way, you are externalizing information and thus reducing the load on your working memory, which would presumably result in a better comprehension of the material at hand. In addition, externalized information can be internalized again and interpreted in a rather lightweight cognitive process with reduced cognitive load [Zha97]. In the literature, cognitive offloading have been shown to improve users' performance when conducting different types of tasks [RMCK14, Gil15, GMNKW01, CACS07, CK11]. Likewise, in process modeling, the use of hybrid process artifacts can help to further externalize information about the process. For instance, modeling with the support of a textual annotator (cf. Section 1.4), can help to externalize the mapping between the process specifications and the model elements, which in turn can make the modeling task less cognitively demanding, but also facilitate the comprehension of the model afterwards, since the annotations can also serve as cues for the logic encoded in the model. Another pertinent example is when evaluating whether an execution path is supported in the model. Therein, instead of computing the activities' interplay after each step of execution, one can externalize and offload these computations to a simulation tool.

Chunking and Schemata. Chunking denotes the process of organizing information into chunks [Gra06]. Given the following sequence of characters "DKFRDEIT", it can be retained in different ways. You could, for instance, retain the whole sequence "DKFRDEIT" at once. Another alternative is to separate it into chunks of single characters "D","K", "F", "R", "D", "E","I","T". A third approach, in turn, would be to decompose the sequence into chunks of two characters each "DK", "FR", "DE", "IT". This way, you are likely to notice that each chunk refers to the acronym of a country (e.g., DK refers to Denmark) and thus, you would better retain the sequence of characters and interpret its content with a reduced mental effort. The *pre-knowledge* allowing you to find the right chunking strategy to decompose and aggregate information pieces is referred to in the literature as a schema, which denotes an abstract mental structure allowing to represent generic concepts in memory [Rum17]. Acquiring the appropriate schemata help to easily encode and retrieve information from humans' memory [Kal09, PRS03, Swe88, SC94]. In the process modeling scope, teaching modeling practices (e.g., modeling patterns) can provide users with schemata facilitating the representation and recognition of specific behaviors and patterns in the model.

All in all, different approaches can be adopted to relieve the load on users' working memory during process modeling and comprehension tasks. The investigated hybrid process artifact and modeling practices could presumably support these approaches and hence improve users' experience with dealing with declarative process models.

CHAPTER 3

Hybrid Business Process Representations: A Conceptual Framework and Systematic Literature Review

---

This chapter summarizes **Article 1** (cf. Part II, Chapter 7):

- **On the declarative paradigm in hybrid business process representations: A conceptual framework and a systematic literature study.**
  Andaloussi, A. A., Burattin, A., Slaats, T., Kindler, E., & Weber, B.
  Information Systems [ABS$^+$20], 2020.

Figure 3.1 highlights the scope of this chapter (and the underlying article). In this contribution, hybrid process artifacts are conceptualized within hybrid business process representations, which in turn are investigated in an SLR (Systematic Literature Review). Section 3.1 describes the context and the moti-

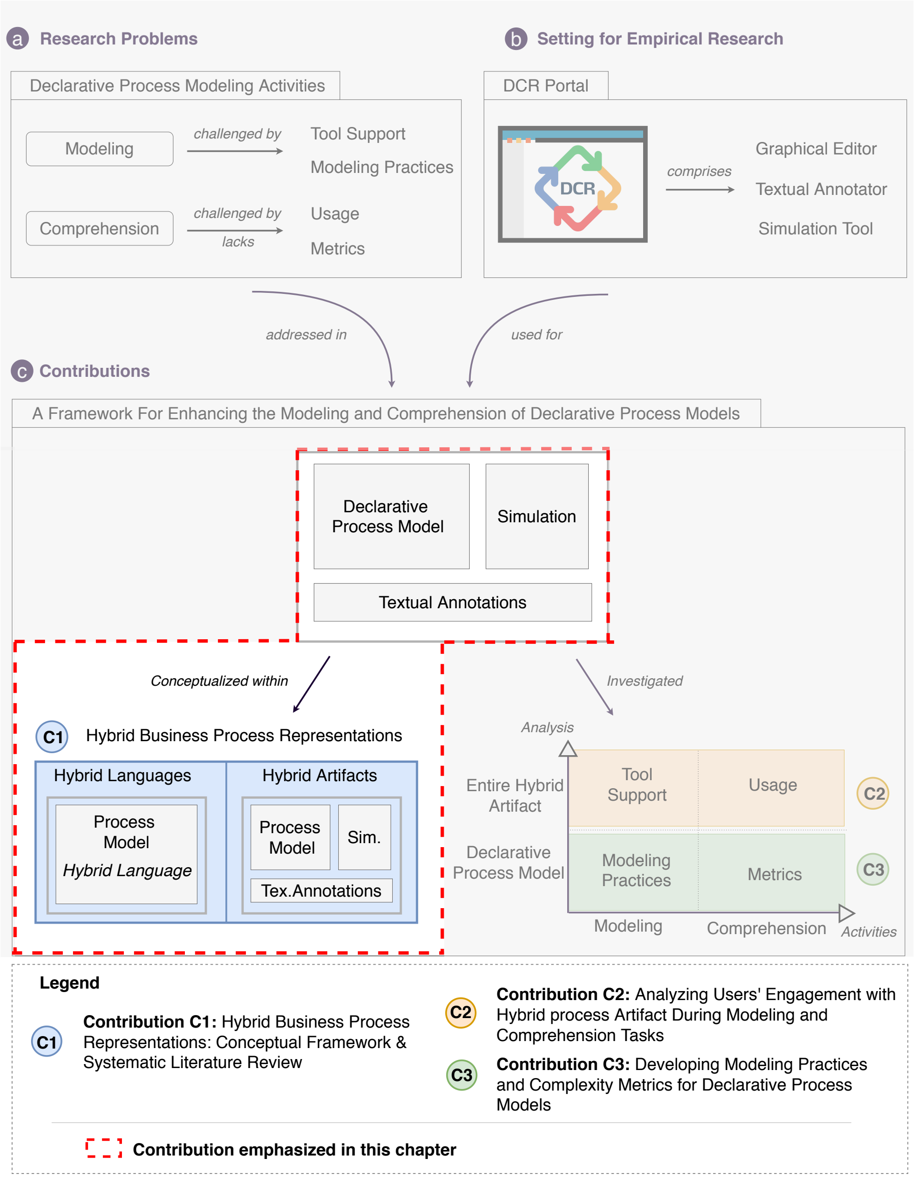


Figure 3.1: Chapter 3 scope.

vation for this contribution, Section 3.2 summarizes the proposed conceptual framework and Section 3.3 provides an overview on the results of the SLR. The aspects discussed in this chapter are presented in detail in Article 1 [ABS^{+}20] (cf. Chapter 7).

3.1 Context and Motivation

The research summarized in this chapter represents an initial stepping stone towards the investigation of the artifacts introduced in the DCR Portal. As mentioned in Section 1.4, the portal includes three design artifacts, i.e., a graphical editor, a textual annotator and a simulation tool. These three artifacts together provide a *hybrid* representation where declarative process models are interwoven with textual annotations and guided simulations to support different modeling and comprehension tasks. Such a hybrid representation is, in fact, not unique to the DCR platform. In the literature, similar representations have been proposed (e.g., [LDHM18, KN19, DSDWSV18, WISW17, SSMR16, DSDWSV16, MSS16, Hin16, DSDWVP16, DGDMM15, DHMS18, WS13, MB13, ZPW11]). Although, these representations share many traits, no clear foundations have been set to define and characterize them. As a result, they were introduced from different perspectives and often with different terminologies. Hence, it was difficult to develop a clear overview on the existing hybrid representations and discern how they reassemble and differ from each other.

Looking at some of the existing hybrid representations (e.g., [SSMR16,DHMS18]), it emerged that the hybrid feature comes at different levels of these representations. In [SSMR16], for instance, the term “hybrid” was used to indicate imperative and declarative languages combined into a *hybrid language*, while in [DHMS18], the terminology was rather used to designate a process model and guided simulations combined into a *hybrid process artifact*. Similar to [DHMS18], the combination of the (DCR Portal) artifacts, investigated in this thesis, can be seen as a *hybrid process artifact*.

The end-goal of this work, however, was beyond the distinction between hybrid languages and hybrid process artifacts. Indeed, the aim was to understand what concepts clearly define these representations and how they have been introduced in the literature. Answering these questions, in turn, required a conceptual framework, setting the foundations for hybrid process representations, and an SLR exploring the different contexts, motivations and characteristics associated with the different representations proposed in the literature. These needs were addressed in this contribution. The following sections (Section 3.2 and Section 3.3) provide an overview of the conceptual framework and summarize the SLR findings respectively.

3.2 A Conceptual Framework for Hybrid Representations (Article 1)

This section summarizes the key concepts defining a process artifact. These concepts are used to differentiate hybrid languages and hybrid process artifacts.

Figure 3.2(a) depicts a conceptual framework defining a *process artifact* in relation to different concepts, represented as distinct components in the framework (i.e., *business process*, *mental model*, *conceptualization*, *language*). In this framework, a process artifact can be seen as an *external representation* of a *business process* delineating the way the process operates in the real-world [AKR07]. Moreover, a process artifact can be considered as a *reflection* of a modeler's *mental model*, that is, in turn, an *internal (mental) representation* capturing his (own) understanding of the business process [SKW11].

The core BPM concepts (i.e., process, case, task, activity) and process aspects (e.g., control-flow, organization, information) [AKR07] can be represented as a *conceptualization* entity within this framework. This entity structures the modeler's mental model by providing a set of concepts and ontologies, allowing him to develop the appropriate mental structures for abstracting the business process [SKW11, Zug13]. In addition, the conceptualization entity gives structure to the process artifact through its underlying process modeling concepts and ontologies [AKR07].

The *language* entity provides the last puzzle-piece for the process artifact framework. On the one hand, it allows to express and communicate the process artifact. On the other hand, it enables to document and communicate the concepts and ontologies within the conceptualization entity, and thus serves as an instrument to transfer the BPM knowledge and make it accessible to others [Gui05].

The proposed conceptual framework can be instantiated in different ways. In Figure 3.2(b) two instances (marked with different colors) are illustrated: a *hybrid language* instance and a *hybrid process artifact* instance. Both instances use the aforementioned components and adhere to the described interplay. The variations come mainly within the *language* and *process artifact* components, which are described in the following paragraphs.

Hybrid Language Instance. In this instance (i.e., marked in blue in Figure 3.2b), the language component comprises a hybrid language that combines the vocabulary of different imperative and declarative languages, while the artifact component comprises a single process artifact expressed using a hybrid language.

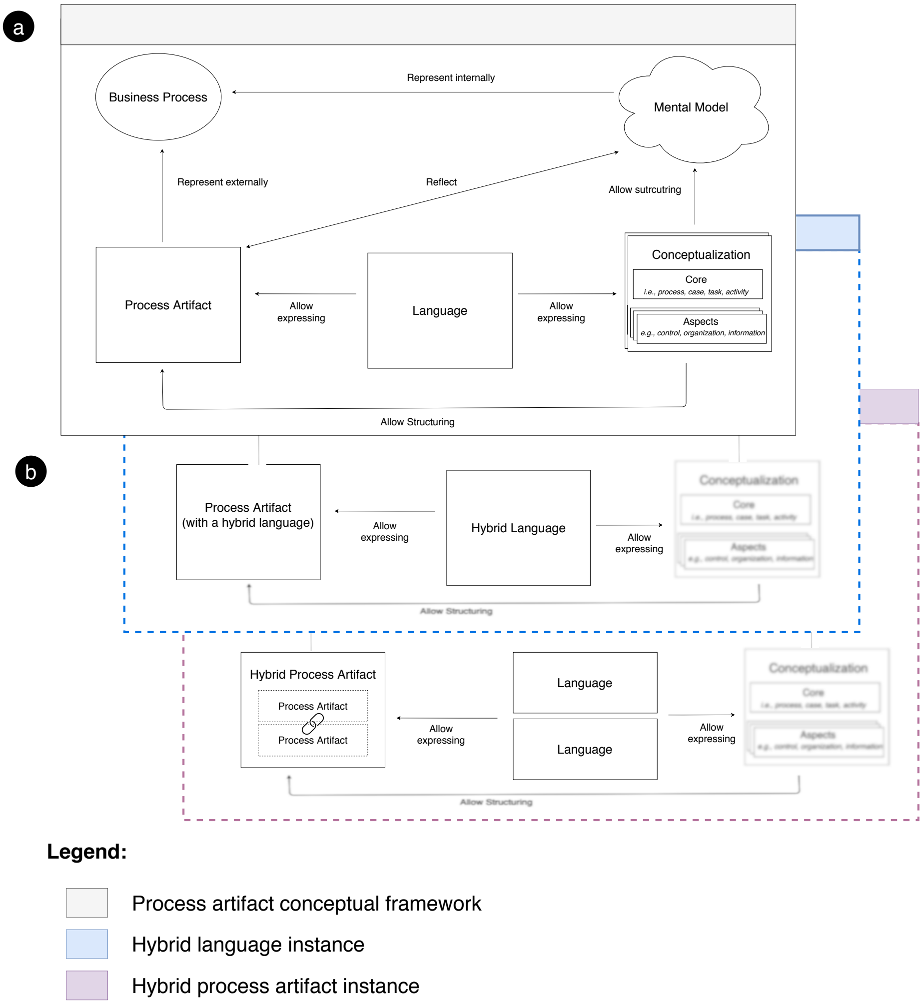


Figure 3.2: A Conceptual framework for process artifacts and a set of possible instantiations of this framework (marked with blue and purple colors). Adapted from [ABS$^+$20] (Article 1).

Hybrid Process Artifact Instance. In this instance (i.e., marked in purple in Figure 3.2b), the language component can comprise two (or many) languages that are used separately to express different process artifacts, which when combined provide a hybrid process artifact (represented in the artifact component).

This conceptual framework provides a unified terminology supported by a set of conceptual bases, which can be used by researchers to better characterize their hybrid representations and position their contributions. Moreover, in the context of this thesis, the conceptual framework served as a basis for guiding the SLR search and organizing the findings, which in turn allowed to delve further into the characteristics of hybrid representations (cf. Section 3.3).

3.3 A Systematic Literature Review About the Declarative Paradigm in Hybrid Representations (Article 1)

This section provides an overview of the findings of the SLR study. As mentioned in Section 1.5, the literature review emphasizes the declarative paradigm in HBPRs (Hybrid Business Process Representations), meaning that all covered studies include an artifact expressed partially or fully using a declarative language.

The SLR addresses a series of research questions aiming at (a) identifying the existing HBPRs, (b) discerning the research lines where they have emerged, (c) investigating the motivations behind their emergence, (d) examining the combined languages and artifacts, (e) developing a taxonomy for hybrid representations (f) evaluating their maturity and (g) identifying the domains where they are applicable. The following paragraphs provide a brief overview of these aspects and highlight some pertinent insights from the research agenda developed based on the findings of the literature review.

Following a systematic approach for searching and reviewing the literature [KC07], 30 articles were identified and examined. Overall, the articles were published in different conference and journal venues in an increasing trend over the last two decades.

The analysis of the identified articles shows that HBPRs have emerged into two different research lines. In the first research line (shortly, *RL1*), hybrid languages (e.g., [SSMR16, SSO01, MS02, WS13, DSDWVP16]) were proposed to represent the rigid and flexible parts of the business process in the same process artifact, while hybrid process artifacts (e.g., [LDHM18, ZPW11, ZPW12, DSDWSV16]) were proposed to support modelers when performing different activities on declarative process models. Regarding the second research line (shortly, *RL2*) hybrid process artifacts (e.g., [KKN12, KN19, CSI11, WISW17]) were suggested to improve the integration of business rules in process models. To this end, authors have recommended to extract business rules from the process (general) control-flow and represent them in hybrid artifacts expressed in rule-based (e.g., Semantics of Business Vocabulary and Business Rules – SBVR [Man17]) or natural languages.

The HBPRs proposed within both research lines share several motivations, notably, the need to support process modelers during modeling, maintainability and comprehension tasks (e.g., [ZPW11, DSDWSV16, WISW17]) as well as to ensure better flexibility and adaptability at run-time [SSMR16,SSO01,MGR11]. Besides, many of the studies in RL1 highlighted the need for concise and precise languages, allowing to model business processes that include both rigid and flexible parts (e.g., [WS13, DGDMM15]), while the studies in RL2 were driven by the need to integrate business rules in a manner allowing them to be defined and changed without affecting the general control-flow of the process (e.g., [VEIP08, GBS07]).

The HBPRs proposed in both research lines use different languages and artifacts. With regards to hybrid languages, Declare [PSVdA07] was the most recurrent language that has been combined with different imperative and declarative languages (e.g., [WS13, SSMR16]). As for hybrid process artifacts, Declare [PSVdA07] and DCR [HM11] were commonly used to express process models that have been extended with textual annotations and guided simulations, whereas, BPMN [OMG06] was mostly used in artifacts that have been integrated with business rules (e.g., [KKN12, CSI11]) that were typically represented in textual formats.

In addition, the results of the SLR allowed to delve further and explore the different types of hybrid languages and hybrid process artifacts. Figure 3.3 summarizes a taxonomy to further characterize hybrid representations. Herein, hybrid languages can be organized into languages providing models with a *hierarchical structure* (e.g., [SSMR16,SSO01]) and languages providing models with a *mixed structure* (e.g., [WS13, DGDMM15]). In the former set of languages, process models are fragmented into several sub-processes, each expressed using a particular language, whereas in the latter set, languages are combined within the same sub-process (or process). Hybrid process artifacts, in turn, can be organized into those combining a process model with textual annotations (e.g., [KKN12, CSI11]) and those combining a process model with guided simulations (e.g., [ZPW12, DHMS18]).

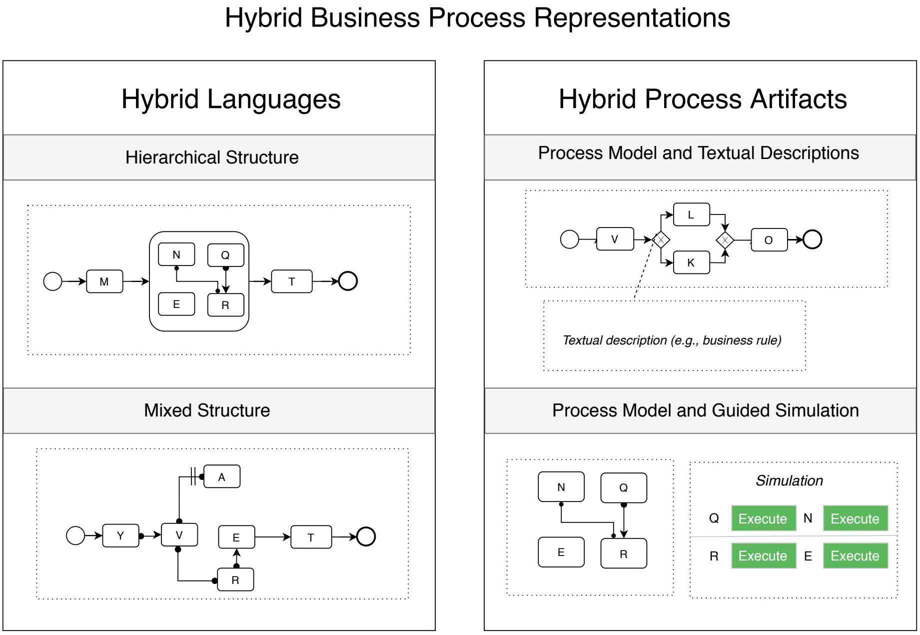


Figure 3.3: A taxonomy showing the different types of hybrid languages and hybrid process artifacts. In the depicted models, normal arrows are used to illustrate the syntax of imperative languages (e.g., BPMN [OMG06]), while the other types of arrows are used to illustrate the syntax of declarative languages (e.g., Declare [PSVdA07]). Adapted from [ABS^{+}20] (Article 1).

The approaches proposed in the literature have been evaluated in terms of their maturity, which has been assessed based on their formalization, availability of implementations and empirical evaluations. Overall, most of the formalized approaches addressed hybrid languages, whereas most of the implementations and empirical evaluations covered hybrid process artifacts. Last but not least, HBPRs have been proposed to model business processes in several domains, notably health care (e.g., [SvS10]), education (e.g., [Hin16]) and Customer Relation Management (e.g., [MS02]).

The analysis of the literature studies was followed by a research agenda for the community. The agenda marks the gaps in the literature and traces out pertinent directions for future work. It addresses aspects related to the design, modeling and evaluation of hybrid language and hybrid process artifacts. With regards to the evaluation of hybrid process artifacts (in relation to this thesis), their importance in supporting declarative models is largely claimed in the literature. Nevertheless, although hybrid process artifacts have been more investigated compared to hybrid languages, the amount of empirical studies is still relatively small compared to those investigating single process artifacts in the literature [Fig17]. Moreover, little is known about the way users engage with hybrid process representations during comprehension and process modeling tasks [ABS$^+$20]. Except for [ZHPW13] where hybrid process artifacts were investigated for their communication support during process modeling, most of the other studies (e.g., [WISW17, DSDWSV18]) focused on performance measures (e.g., accuracy, response time). In turn, less attention was devoted to understand the dynamics underlying users' visual behavior and interactions with hybrid process artifacts. Therefore, the opportunities and challenges associated with the use of hybrid process artifact remain still unclear in the literature.

Overall, the outcome of the SLR complements the proposed conceptual framework and delivers an overarching understanding of the state of the art literature about HBPRs. This outcome is expected to support the development of new hybrid representations and provide a base that can be systematically updated with the new research.

CHAPTER 4

Analyzing Users' Engagement with Hybrid process Artifact During Modeling and Comprehension Tasks

---

This chapter summarizes **Articles 2 and 3** (cf. Part II, Chapters 8 and 9 respectively):

- **Exploring the modeling of declarative processes using a hybrid approach.** (Article 2)
  Andaloussi, A. A., Buch-Lorentsen, J., López, H. A., Slaats, T., & Weber, B.
  International Conference on Conceptual Modeling [ABLL^{+}19], 2019.
- **Exploring how users engage with hybrid process artifacts based on declarative process models: a behavioral analysis based on eye-tracking and think-aloud.** (Article 3)

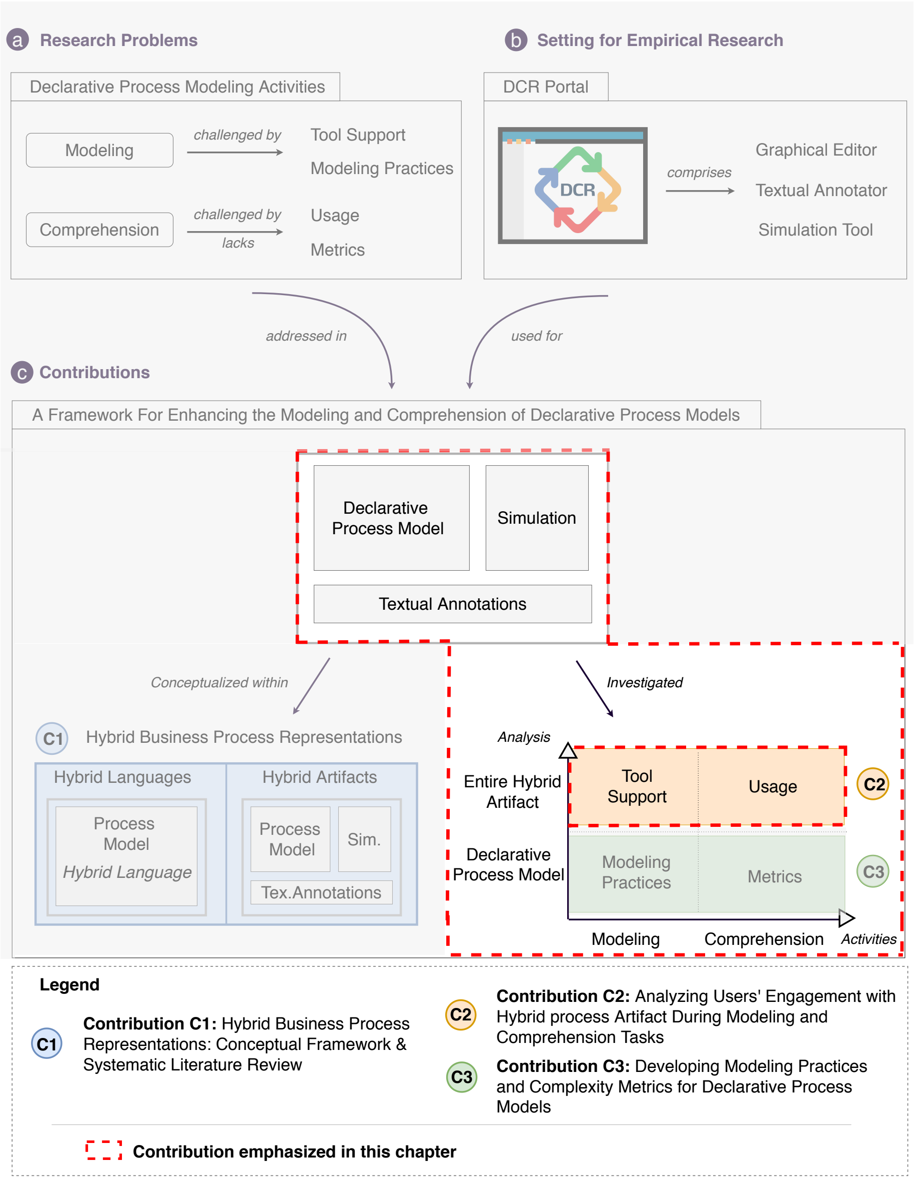


Figure 4.1: Chapter 4 scope.

Andaloussi, A. A., Zerbato, F., Burattin, A., Slaats, T., Hildebrandt, T. T., & Weber
Software and Systems Modeling [AZB$^+$20], 2020.

Figure 4.1 highlights the scope of this chapter (and the underlying articles). The research conducted in this contribution comprises two studies: the first study addresses the quality of the tool-support, provided by the DCR hybrid process artifact during process modeling, while the second study addresses the usage of the hybrid artifact during model comprehension. The analysis carried-out in this contribution investigates the entire DCR hybrid process artifact. Section 4.1 describes the context and the motivation for this contribution, Section 4.2 summarizes the study investigating process modeling with the support of hybrid process artifacts, while Section 4.3 provides an overview of the study investigating the comprehension of process models embedded in hybrid process artifacts.

The insights about process modeling and model comprehension using hybrid process artifacts are presented in detail in Article 2 [ABLL$^+$19] (cf. Chapter 8) and Article 3 [AZB$^+$20] (cf. Chapter 9) respectively.

4.1 Context and Motivation

The research summarized in this chapter aims at investigating the use of hybrid process artifacts during process modeling and model comprehension tasks. As mentioned in Section 1.3, the modeling and comprehension of declarative process models are associated with a set of challenges. Indeed, it is difficult to keep track of the mapping between the process specifications and the elements in the process model. To address this limitation, an explicit mapping between the information exposed by both artifacts is required. Another limitation is that declarative process models do not explicitly depict the execution paths allowed in the model. Subsequently, to verify whether the process model supports a certain execution path, users are required to internally compute the interplay between the process activities after each execution step, which, in turn, increases the load on their working memory (cf. Section 2.2.1).

In the literature, a few hybrid process artifacts have been proposed to support process modeling using declarative languages [ZPW11, LDHM18]. However, existing evaluations have covered only the support of guided simulations during process modeling [ZHPW13]. Conversely, little is known about users' behavior when modeling declaratively using a hybrid artifact enriched with a textual annotation tool. This, in turn, raises the need to explore this angle and investigate the potential support of this artifact.

With respect to the comprehension of declarative process models, existing studies have addressed the support offered by hybrid process artifacts from a user performance perspective (e.g., [WISW17,DSDWSV18]), whereas the actual use of these artifacts during comprehension tasks was not clearly covered. Investigating users' behavior when engaging with hybrid process artifacts is important to understand how declarative process models are used – but also how they are combined with the other artifacts of the hybrid representation during comprehension tasks. Addressing these questions is needed for tailoring existing process representations and tool-support to the individual need of users and for improving the effectiveness of existing modeling approaches. The following sections (Section 4.2 and Section 4.3) summarizes the studies investigating the use of hybrid process artifacts in modeling and comprehension tasks.

4.2 Analyzing Process Modeling Using Hybrid Artifacts (Article 2)

This section summarizes the study exploring process modeling with the support of hybrid process artifacts (Article 2 [ABLL$^+$19], cf. Chapter 8).

To evaluate the way modelers actually use the DCR hybrid process artifact during process modeling tasks, a qualitative study was designed and conducted with seventeen novice modelers, including students with a technical background and employees at a Danish municipality. The study focused primarily on the role of the textual annotator (called the Highlighter in Article 2 [ABLL$^+$19]) within the hybrid representation. It addressed two key research questions aiming at (a) understanding how users engage with the hybrid process artifact during process modeling tasks and (b) identifying the aspects where the textual annotator, within the hybrid representation, could contribute to enhancing the model quality. To address these aspects, the participants were introduced to the hybrid process artifact and then given a process modeling task. By the end of the task, the participants were asked to reflect on their experience in retrospective think-aloud sessions [HNA$^+$11]. During the analysis, modelers' interactions with the hybrid process artifact were examined to identify the way the different artifacts were used during the modeling task. In addition, modelers' verbal data was analyzed following a qualitative approach based on concepts from grounded theory [CB07].

Table 4.1 summarizes these findings in terms of the use of the hybrid artifact and the presumable influence of the textual annotator on the quality of process models. These aspects are further discussed in the following paragraphs.

Regarding the use of the hybrid process artifact, the analysis of modelers' interactions showed that the textual annotator was mainly used at the beginning

CATEGORIES	INSIGHTS
Use of the hybrid process artifact	• The textual annotator is used to identify activities and roles • The graphical editor is used to add relations to the process model
Influence of the textual annotator on model quality	• Supports better traceability and greater coverage of the process specifications • Helps to evaluate the alignment between the process description and the produced process model • Provides better means to document the model semantics and keep track of modelers' design decisions

Table 4.1: A summary about the use of the hybrid artifact and the presumable influence of the textual annotator on the quality of process models. Based on the insights presented in [ABLL$^+$19].

of the modeling session to identify activities and roles whereas most of the constraints were directly modeled (as DCR relations) in the graphical editor. By trying to further substantiate this pattern from participants' verbal utterances, it has emerged that the textual annotator served as a kick-start for the process modeling task. As explained by some participants, the annotation of activities and roles in the text helped them to develop a quick overview of the process. Moreover, the annotations were perceived to provide structure and help to decompose the process description into smaller fragments. Furthermore, some participants affirmed that, by annotating the text, they could better memorize and easily identify the process specifications. The participants also justified their abstention from using the textual annotator to mark constraints in the text. Therein, they preferred a graphical (two-dimensional) visualization, where it was easy to perceive the interplay between the process activities. In addition, the participants have reported struggling to identify explicit constraints in the text that can be easily mapped to the different DCR relations. These insights suggest that the textual annotator provides a higher cognitive fit for annotating activities and roles, while the graphical editor gives better cognitive fit for identifying the DCR relations.

With respect to the presumable influence of the textual annotator on the quality of the produced process models, during the retrospective think-aloud sessions, modelers have affirmed that the tool, when used as part of the hybrid artifact, could support better traceability and greater coverage of the process specifications described in the text. In addition, some participants affirmed using the textual annotator together with the graphical editor to evaluate the alignment between the process description and the produced process model by the end of the modeling task. Last but not least, participants have also emphasized the use of the textual annotator to document the model semantics and keep track of modelers' design decisions.

All in all, the outcome of this study reveals the use of hybrid process artifacts during process modeling. Herein, the results suggest that these representations can provide a convenient modeling tool-support and can presumably help to produce process models with enhanced quality. However, finding an explicit mapping between the process specifications and the model elements can be challenging using a textual annotator. These insights pave the way for improved modeling tool-support. For instance, in the context of the DCR platform, the findings of the study served as a basis to improve the quality of the process annotator by proposing a new version, where natural language processing (NLP) was used to automate the identification and mapping of constraints in the process specifications [LMMS19].

4.3 Analyzing Model Comprehension Using Hybrid Artifacts (Article 3)

This section summarizes the study exploring model comprehension with the support of hybrid process artifacts (Article 3 [AZB$^+$20], cf. Chapter 9).

The investigated hybrid artifact (called DCR-HR in Article 3 [AZB$^+$20]) is similar to the one used for modeling tasks. It comprises a process model represented in the DCR language, textual annotations, mapping the process specifications with the model elements, and guided simulations allowing to illustrate the model behavior. The hybrid process artifact is meant to support users with different backgrounds when performing comprehension tasks requiring to extract different types of information about the process. These tasks could, for instance, include (1) asking about specific constraints in the model, (2) prompting the users to decide among a set of options allowed in the model or (3) determining whether a course of events is compliant with the process model. The tasks were given in the context of a real-world process regulated by the consolidation act on social services, i.e., a law used in the Danish local government to administer the rights to benefits for young persons with special needs [The15]. This process was modeled as a DCR Graph, which was, in turn, embedded in a hybrid process artifact where the model elements were mapped with textual annotations referring to excerpts of the law text.

To evaluate the way users approach the different tasks, a qualitative study was designed and performed with fifteen participants, including employees at a municipality in Denmark and academics working or studying at Danish universities. These two profiles served as proxies for domain experts and IT specialist who are usually involved in the development of such processes. The study addressed three research questions aiming at (a) understanding how the hybrid process artifact is used to conduct different tasks (b) identifying the benefits and challenges associated with each artifact of the hybrid representation and (c) delineating the common strategies adopted by the participants when searching information in the hybrid artifact. To answer these questions, the participants were, first, familiarized with the hybrid process artifact, then given a set of comprehension tasks and finally asked in retrospective think-aloud sessions to verbalize their thoughts and reflect on their experience with the hybrid representation [HNA$^+$11].

The analysis covered both eye-tracking and verbal data. With regards to the former, participants' eye movements were analyzed at different levels of granularity. The coarse-grained analysis was meant to study the distribution of attention among the different artifacts and scrutinize the way these artifacts were combined to perform different tasks. To this end, process mining techniques [VDA16] were used to discover the processes guiding users' eye movements on the hybrid representation. These (behavioral) processes were, in turn, analyzed to derive the common reading patterns exhibited by the participants when solving different tasks. The outcome of the analysis allowed to compare the behavior of municipal employees and academics, as well as to evaluate the influence of different task types.

The fine-grained analysis, in turn, aimed at studying the temporal patterns in the eye-tracking data. While the previous investigation remained at the artifact level, this analysis delved into the level of the model activities. Herein, the sequence of participants' gazes captured using eye-tracking were unfolded along a timeline and were appended qualitative memos describing the observed gazing patterns. These memos were, subsequently, analyzed following a qualitative approach based on grounded theory [CB07]. The outcome of the analysis allowed to discern the common search strategies adopted by the participants when engaging with the hybrid representation.

The analysis of verbal data was intended to provide a deeper understanding of participants' behavior and give subjective insights about their experience when dealing with the hybrid process artifact. This analysis was conducted following a qualitative approach guided by a set of coding techniques based on grounded theory [CB07]. The obtained verbal utterances were, subsequently, triangulated

ARTIFACTS	BENEFITS	CHALLENGES
Process Model	• Provides an overview on the process • Helps to navigate in the law text	• The model semantics are challenging (**ME**) • The model may not capture as much as details as the law text
Law Text (*shown in the textual annotations*)	• More comprehensive than the model • Supports better decision-making (**ME**)	• The linguistic patterns are hard to read (**AC**)
Simulation	• Clarifies the semantics of the model • Allows illustrating different execution paths • Allows validating users' assumptions on the model	• Inefficient for evaluating all the possible executions paths in the model (**AC**) • Can be time-consuming (**ME**)

Table 4.2: Challenges and benefits associated with different process artifacts. The insights were perceived by two groups of participants: municipal employees (shortly, ME) and academics (shortly, AC). The insights labeled with ME and AC in the table were particularly perceived by the participants of the respective group, while the other insights were common to both groups. Adapted from [AZB+20].

with the eye-tracking data (cf. Section 1.6.1), which, as a result, provided a better interpretation of the observed eye-tracking behaviors and revealed the benefits and challenges perceived by municipal employees and academics when performing different tasks.

The outcome of the analysis provided informative insights. With regards to the use of the hybrid representation, it has emerged, from the coarse grained-analysis of the eye-tracking data, that different groups of stakeholders combine different artifacts and that their usage changes depending on the type of the task being performed. With respect to the participants' background, municipal employees were more reliant on the textual annotations, while academics focused more on the process model. Participants' verbal utterances showed that these patterns can be associated with the lack of proficiency in reading process models and legal texts reported respectively by municipal employees and academics. These observations suggest that a hybrid process artifact is indeed important to provide a unified process representation that can be interpreted by different stakeholders. As for the task type, the participants relied on different artifacts depending on the nature of the information prompted by the task. For example, when being asked to determine whether a course of events is allowed in the model, the guided simulations provided better cognitive fit (cf. Section 2.2.2) with the task and helped participants to offload the computation of the interplay between the process activities at different stages of execution. Likewise, for the other types of tasks, either the process model or the textual annotations created a better cognitive fit. These insights remind the importance of providing hybrid representations combining different artifacts to ensure better cognitive fit depending on the task type.

With respect to the benefits and challenges associated with the different artifacts of the hybrid representation, some aspects were common between both municipal employees and academics, while other aspects were subject to the participant background. Table 4.2 provides a summary of these aspects. Notably, the process model provided a good overview of the process for both municipal employees and academics. Although municipal employees were challenged by the formal semantics of the process model, they could still refer to the mapping between the model elements and the textual annotations (referring to excerpts of the law text) to navigate between the different parts of the process. The law text in the textual annotations, in turn, was particularly challenging for academics who were unable to understand the linguistic patterns encoded in the law. As for the simulation, it helped to clarify the semantics of the process model and evaluate different execution paths. However, it was perceived inefficient by some academics and time-consuming by other municipal employees. These insights suggest that each artifact is associated with a set of benefits and challenges, which are differently perceived by stakeholders. Using a hybrid process artifact supporting the dual-coding of information, in turn, could provide stakeholders with additional channels to meet their individual preferences and improve their understanding of the process (cf. Section 2.2.2).

The last part of the analysis focused on the search strategies adopted by the participants when engaging with the hybrid representation. The findings discerned a *goal-directed* strategy characterized by long gazes on the task-relevant elements in the model, and a *exploratory* strategy characterized by short and rather random gazes mainly on the non-relevant part of the model. Moreover, it was observed that participants tend to switch from an exploratory to a goal-directed strategy as time goes on. The analysis of the participants' search strategies has also revealed important insights about the use of the guided simulations. Therein, two different patterns were observed: a first pattern where the simulation was intertwined with the process model throughout the whole task and a second pattern where the simulation was rather used towards the end of the task. The verbal utterances suggest that the former pattern corresponds to a strategy adopted by participants who relied on the simulation to understand the whole model, while the latter strategy was used to validate or confirm one's interpretation of a certain specification depicted in the model or described in the law text. These insights show that participants adopted different search strategies and used the simulation tool in different manners.

All in all, the outcome of this study delineates the usage of hybrid process artifacts during model comprehension tasks and motivates their adoption in practice. In addition, it paves the path for the development of new hybrid process artifacts, providing context-adaptive support to users based on their background and the nature of the task in hand. These hybrid artifacts can adjust to users' preferences based on their explicit feedback or implicit behavior.

CHAPTER 5

Developing Modeling Practices and Complexity Metrics for Declarative Process Models

---

This chapter summarizes **Articles 4 and 5** (cf. Part II, Chapters 10 and 11 respectively):

- **Understanding quality in declarative process modeling through the mental models of experts.** (Article 4)
  Andaloussi, A. A., Davis, C. J., Burattin, A., López, H. A., Slaats, T., & Weber, B.
  International Conference on Business Process Management [ADB+20], 2020.
- **Assessing the complexity of declarative process models using model-based metrics.** (Article 5)
  Andaloussi, A. A., Burattin, A., Slaats, T., Kindler, E., & Weber, B.
  (Reported in the thesis)

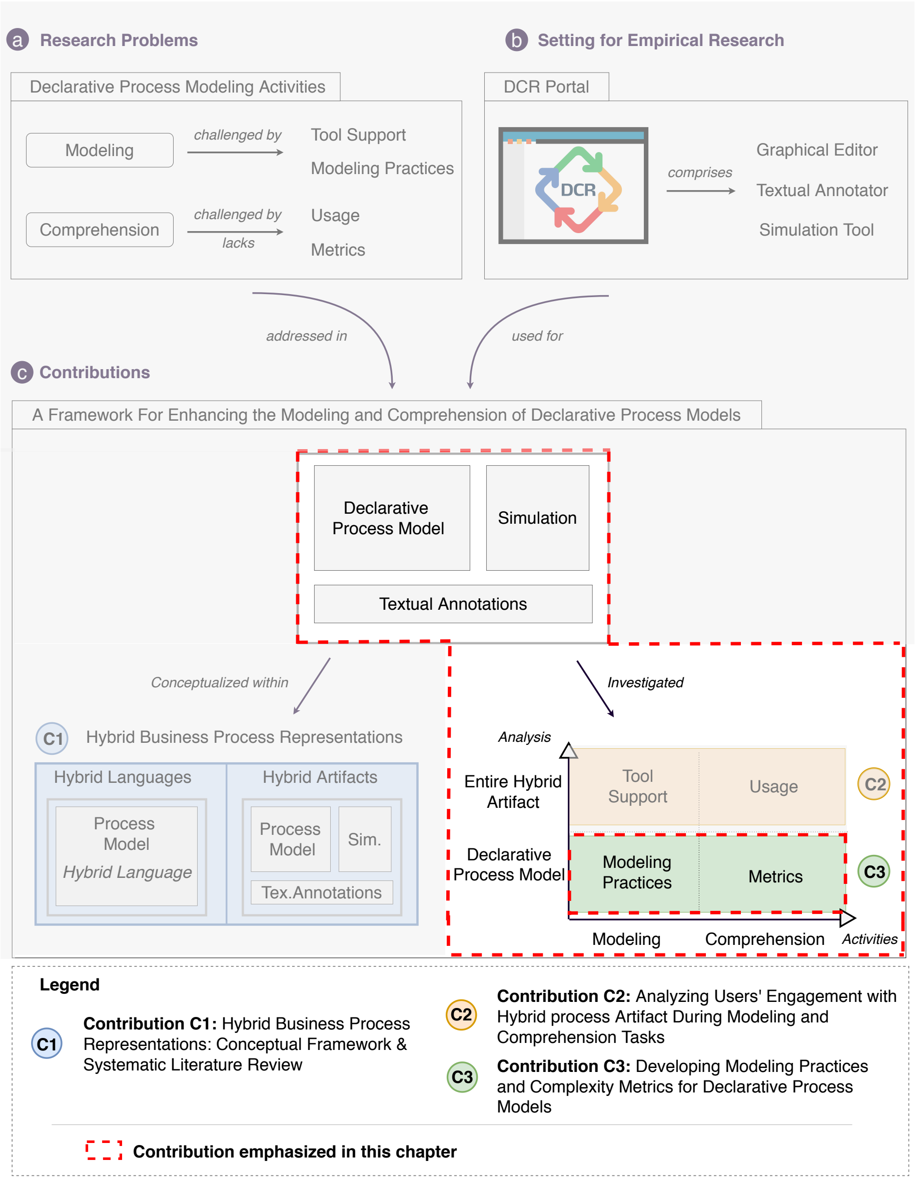


Figure 5.1: Chapter 5 scope.

Figure 5.1 highlights the scope of this chapter (and the underlying articles). The carried-out research comprises two studies emphasizing declarative process models. The first study investigates the modeling practices guiding process modeling tasks, while the second study develops and evaluates a set of complexity metrics allowing to appraise the comprehension of declarative process models. Section 5.1 describes the context and the motivation for this research, Section 5.2 summarizes the study investigating declarative modeling practices, while Section 5.3 provides an overview about the study presenting and evaluating complexity metrics for declarative process models. The insights reported in these two sections are presented in more detail in Article 4 [ADB$^+$20] (cf. Chapter 10) and Article 5 (cf. Chapter 11) respectively.

5.1 Context and Motivation

The work summarized in this chapter aims at inferring a set of modeling practices and complexity metrics. The former are meant to guide modelers when designing declarative process models, while the latter are intended to provide quantifiable means to assess the understandability of declarative models. Modeling practices are central to the modeling of business processes. These practices are taught from the guidelines and recommendations delineating how to better conceptualize and represent business processes. While this topic has been largely investigated in the literature [Kro16, MRvdA10, RMR10, CFF$^+$18], most of the proposed guidelines focus on languages from the imperative paradigm. By looking, for instance, at the 50 imperative process modeling guidelines summarized by Corradini et al. [CFF$^+$18], one can notice that several guidelines may not apply to declarative models. Therein, several questions could arise, for example, “Should declarative process models have a single start event?”, “Does it make sense to reduce concurrency and sequentialize a declarative model like an imperative one?” or “Should one organize the layout of declarative and imperative process models in the same way?”. These questions represent only a subset of those which one might wonder when trying to apply imperative modeling guidelines to declarative models. To demystify these and similar issues, it is important to investigate the aspects defining the quality of declarative models and propose a set of practices to guide modelers during process modeling.

Another key challenge motivating the research summarized in this chapter is the lack of complexity metrics for declarative process models. As mentioned in Section 1.3.2, while there is a large interest in model-based metrics [Men07], the existing studies focus only on imperative process models. Therefore, in the current state of the art, it is still hard to objectively decide when a declarative model is easy to comprehend or rather needs some improvements. However, despite the differences between imperative and declarative languages, they are both based on graphical representations taking roots in the graph theory and therefore, it is probable that, with some adjustments, a subset of existing metrics could become applicable to declarative process models as well. To investigate this proposition, it is important to conduct further research in this direction. The following sections (Sections 5.2 and 5.3) summarize the studies investigating process modeling practices and complexity metrics.

5.2 Declarative Modeling Practices (Article 4)

This section summarizes the study investigating declarative modeling practices (Article 4 [ADB^{+}20] cf. Chapter 10). These practices originate from a set of *quality dimensions* which represent the *scales* based on which experts evaluate the quality of declarative process models in practice.

Eliciting experts' quality dimensions enables to externalize their knowledge and thus make the criteria by which declarative models are judged more tractable. Based on that, it becomes possible to develop a catalog of modeling practices to guide novices in declarative process modeling.

In this work, Personal Construct Theory (PCT) [Kel55] provided the theoretical framework to elicit experts' quality dimensions. The theory posits that individuals develop a set of interrelated *personal constructs* (i.e., dimensions), emerging from their past and ongoing experiences [Kel55]. Articulating these dimensions is usually guided by the Repertory Grid approach [Kel55,SSF81,TH02] i.e., a knowledge elicitation procedure helping individuals to verbalize their own personal constructs. PCT and Repertory Grid have been applied in clinical psychology [Kel55], market research [ML00] and technology acceptance studies [DH07]. However, their potential has not been exploited in process modeling studies. To unleash this potential, the theory and the underlying approach were adjusted to fit the purpose of this study. Thereafter, a step-wise approach was documented and used to infer experts' quality dimensions and based on that, build a catalog of modeling practices.

The study has been conducted in two phases. The first phase consisted of designing a process modeling task where thirteen novices, intermediate and expert modelers were given a set of process specifications describing a loan-application process [ETvdHP16] and asked to translate it into a declarative process model in the DCR language. The heterogeneity of the participants' group aimed at exploring a wide range of model complexities and providing models with different qualities. The collected models were, in turn, used in the second phase. Therein, a series of consultations were organized with four experts in declarative process modeling, who, following the adjusted Repertory Grid approach, were able to articulate their quality dimensions and explain the underlying rationals for designing high-quality models. The consultations were recorded and then analyzed following a qualitative approach based on grounded theory [CB07] to formulate a set of modeling practices meant to guide novices when modeling declaratively. Table 5.1 presents a set of categories, themes and quality dimensions, while the next paragraphs provide examples about some of the underlying modeling practices. All findings are presented in more detail in [ADB$^+$20].

The analysis resulted in the identification of twenty-five personal constructs, organized into seven themes and two categories. Experts associated these constructs with a set of modeling practices that can help to improve the quality of declarative process models. Overall, the modeling practices addressed the *semantic* and the *pragmatic* qualities of process models. *Semantic quality* refers to the extent to which a model can make true statements about the process it represents [RMR10]. This category included four themes: *modeling behavior*, *modeling patterns*, *modeling events*[1] and *modeling data*. The *modeling behavior* theme comprised a set of recommendations about how to appropriately capture and represent the model behavior. Therein, for instance, the experts advised avoiding flow-based modeling, that usually underlies a restrictive behavior, and rather raise concurrency in the model to support better flexibility. The experts also highlighted the importance of capturing the activities that represent the goals of the process and model them as required activities that must be eventually fulfilled during the process execution. Another example relates to the modeling of start-events, where experts advised to evaluate whether non-constrained activities are good candidates for being start-events in the model, if not, then they should be constrained by other activities. The *modeling patterns* theme introduced several mechanisms to represent specific behaviors in more concise and precise manners. For instance, the experts explained the way to treat exceptions and how to model the termination pattern. The *modeling of events* theme focused on the way process activities should be captured and represented in the model. Therein, for example, the experts recommended assigning roles to all activities in order to support better traceability and access control. Moreover, the experts advised against using intermediate events i.e., activities that are not part of the process specifications but rather used to enforce specific behaviors in the model. The *modeling of data* theme comprised recommendations about the proper way to integrate contextual data in the process control flow (cf. Section 2.1.2). For example, the experts highlighted the necessity of choosing variable data types that can easily convey meaning about the kind of data they represent. Moreover, the experts pointed out the effects of local and global data variables (i.e., variables evaluated in data expressions either immediately after being set by data activities or postponed to a later stage of execution) and recommended using local data variables in the model.

---

[1] The terms *event* and *activity* refer to the same concept in DCR. The former term is defined in the formal notation, while the latter term is adopted by the DCR modeling tool.

CATEGORIES	THEMES	QUALITY DIMENSIONS
Semantic Quality	Modeling Behavior	• Comprehensiveness of behavior • Presence of behavioral errors • Flow-based versus declarative modeling • Modeling of required events[1] • Modeling of end-events • Modeling of start-events • Multi-instance processing • Modeling against IT silliness • Purpose of the model
	Modeling Patterns	• Use of standard patterns • Condition-response versus include-exclude patterns • Treatment of exception pattern • Use of termination pattern
	Modeling Events	• Role assignment • Use of intermediate events • Implicitness of events
	Modeling Data	• Encoding decisions explicitly or using data expressions • Appropriate choice of data types for data variables • Local/global effect of data variables
Pragmatic Quality	Model Layout	• Alignment and positioning of elements • grouping of events • Visual conciseness
	Event Layout	• Meaningful naming of events • Verb-object versus noun-based naming of events • Color coding
	Data Layout	• Correspondence between variable names and data events' names

Table 5.1: A summary of the elicited quality dimensions. A detailed description of these quality dimensions in presented in [ADB$^+$20] (Article 4, cf. Chapter 10).

The second category addressed the *pragmatic quality* of declarative process models, which denotes the correspondence between the model and the way it is interpreted by the reader [RMR10]. This category comprised three themes: *model layout*, *event layout* and *data layout*. The *model layout* theme emphasized a set of recommendations to improve the overall model appearance. This, for instance, included the appropriate alignment and positioning of the model elements and the grouping of the process activities sharing the same context into nest activities. The *event layout* theme was rather focused on the internal pragmatics of the activities including, for example, the importance of choosing meaningful names to activities and formulating them in a verb-object or noun-object format depending on their purpose. Lastly, the *data layout* theme addressed the naming of data activities and variables (cf. Section 2.1.2). Therein, experts insisted on keeping a clear visual correspondence between the two data elements.

The quality dimensions and the underlying modeling practices identified by expert share many similarities with imperative process modeling guidelines. This is particularly the case for those related to the pragmatic qualities of process models, whereas, for aspects addressing semantic qualities, some disparities were identified (e.g., the modeling patterns shown in Table 5.1). While the common guidelines could be directly applied to declarative process modeling, the disparate ones raise new research questions, which, in turn, require further empirical investigations to understand the impact of these guidelines on declarative process modeling. A more detailed discussion of the similarities and disparities between the quality dimensions in both paradigms is presented in [ADB+20] (Article 4, cf. Chapter 10).

The outcome of this work has impacts on research, education and industry. In research, the obtained insights contribute to a better understanding of quality in declarative process models. In addition, the proposed adaptation of PCT paves the way for further studies conducting similar interpretative analyses of different process modeling approaches. With regards to education, the reported modeling practices can be taught to modelers to guide them and help them to improve the quality of their models. As for the industry, the quality aspects that can be directly derived from the model might be implemented by tool vendors to automate the assessment of model quality at design time and offer modelers a customized tool-support.

5.3 Complexity Metrics for Declarative Process Models (Article 5)

This section summarizes the work investigating complexity metrics for declarative process models (Article 5, cf. Chapter 11). The research carried-out in this direction was twofold. On the one hand, it provided a set of complexity metrics allowing to assess the understandability of declarative process models, particularly in the DCR language. On the other hand, it comprised an empirical study addressing the impact of the factors captured by the proposed metrics on the difficulty of comprehending declarative process models.

The complexity metrics proposed in this work were derived from an existing set of metrics used for imperative process models in the EPC (Event-driven Process Chain) language [Men07]. At first, the EPC metrics were contrasted with the quality dimensions identified in Article 4 [ADB$^+$20] (cf. Section 10). The results allowed to rule out the (inapplicable) metrics capturing the complexity of the sequence-flow in the model (e.g., concurrency metrics), and rather emphasize those capturing structural features at the graph level (e.g., density) that can serve to estimate the complexity of process models expressed in both imperative and declarative languages. The candidate metrics were researched further in the literature as an attempt to lay out the theoretical foundations allowing to define them in the context of declarative process modeling. As a result, four metrics (i.e., *size*, *density*, *separability* and *relation variability*) were defined in mathematical terms. In addition, several hypotheses have been formulated to verify the impacts of the factors captured by these metrics on the understandability of process models, which was addressed from a human-cognitive perspective.

Overall, twelve hypotheses were tested in the empirical study. The hypotheses address the effect of the four factors (i.e., size, density, separability and relation variability) on three indicators of cognitive load (i.e., perceived difficulty, comprehension accuracy and response time [CZW$^+$16]). The study was designed as a controlled experiment which has covered 16 participants having different levels of expertise in declarative process modeling. Following a within-subject approach, the participants were given a series of comprehension tasks about four sets of process models, each addressing a specific factor. The models within each set were designed to support a pairwise comparison between a *reduced* (low) *factor level*, where the metric capturing the addressed factor returns a low value (e.g., low density), and an *increased* (high) *factor level* where the same metric provides a high value (e.g., high density). The models and the tasks were derived following a systematic approach, where potential confounding factors were identified and mitigated during the design phase of the material. The collected data was analyzed following a pairwise approach, where the two levels within each factor were compared within subjects. This approach was adopted to mitigate personal factors associated with the background and expertise of participants. The analysis covered both descriptive statistics and hypothesis-testing.

The results of the analysis showed the difference between the increased and reduced levels of the investigated factors in terms of the used cognitive load measures. Overall, 10 out of 12 hypotheses were confirmed, suggesting, in turn, that the size, density, separability and relation variability factors influence users' cognitive load during model comprehension tasks. The two remaining hypotheses, addressing the effects of the density and separability factors on comprehension accuracy, could not be verified. However, the statistical tests showed that these factors have, indeed, effects on other estimators of cognitive load (i.e., perceived difficulty and response time). A possible explanation to this difference could be attributed to the design of the experiment as the given tasks did not involve any time constraints and therefore, participants might have had more time to review their answers and check their correctness.

It is also worthwhile mentioning that the study is associated with a set of threats to validity, which can be organized into internal, external, construct and conclusion threats. These threats address the potential limitations of the study and highlight the assumptions made when conducting the experiment. Identifying these threats is important for the interpretability of the results and the planning of any follow-up research in this direction. More insights about the identified threats are provided in Article 5 (cf. Chapter 11).

All in all, the outcome of this work delivers quantifiable means to assess the complexity of declarative process models. In addition, the results of the empirical study show the impacts of the factors captured by these metrics on users' cognitive load, which in turn, highlight the importance of considering them when evaluating the understandability of declarative process models. The proposed metrics can be easily implemented by existing tool vendors (e.g., DCR Solutions) and subsequently used to estimate the complexity of the produced models in the modeling platform. Moreover, these metrics can be used as objective design criteria for comparing existing models in terms of their complexity and support for understandability. In addition, assuming that process models can be designed using different combinations of constructs and modeling patterns, the proposed metrics can serve as a basis for suggesting structural changes that systematically optimize process models in terms of these metrics.

CHAPTER 6

Conclusion

This chapter summarizes this document and delineates the most pertinent directions for future work.

This thesis proposed a framework to support the modeling and comprehension of declarative process models. While declarative languages allow to capture and represent flexible processes concisely, their constraint-based approach can hinder their understandability and act as a barrier challenging their adoption in practice. The research presented in the thesis was motivated by a set of issues challenging the modeling and comprehension of declarative process models. With regards to process modeling, the quality of the existing tool-support and modeling practices were focal to this research. As for model comprehension, pertinent issues covered the usage of declarative process models and the need for metrics to assess their understandability.

The research focused on the DCR language that is a known declarative language supported by a suite of industry-level process modeling tools. While a lot of research and development has been conducted on formal aspects of the language and the technicalities of the proposed tools [SHCV15, Nor18, Muk12], little has been done to explore the use of the language and the support of the tools from a user perspective. This facet was addressed in this thesis based on a rigorous scientific framework grounded in the state of the art literature in process modeling and cognitive psychology.

The carried-out research emphasized three key artifacts of the DCR platform (i.e., declarative process models, textual annotations and guided simulations). These artifacts are, indeed, not unique to DCR, but rather serve as representatives for a broad range of similar artifacts proposed in different combinations to support process modeling and comprehension. In the context of this thesis, these conjoined artifacts formed a so-called "hybrid process artifact".

The contributions of the thesis covered the conceptualization and investigation of hybrid process artifacts. The conceptualization of hybrid process artifacts (i.e., `C1`) was among the three key contributions in this thesis. Therein, a conceptual framework and an SLR (Systematic Literature Review) were proposed. As for the investigation of hybrid process artifacts, two contributions (i.e., `C2` and `C3`) were made. While the former contribution examined the entire hybrid artifact, the latter focused on the declarative model within the hybrid artifact.

In the first contribution (`C1`), hybrid process artifacts were conceived within a general conceptual framework. In this framework, they were defined as a subset of hybrid representations, which comprise both languages and artifacts combined at different levels of abstraction. The outcome of the conceptual framework allowed to demystify these different notions and provided a clear and unified terminology, that can be used in the process modeling community to discern the existing hybrid representations and better position the up-coming studies.

The conceptual framework was followed by an SLR where existing hybrid representations were scrutinized at different levels. The results of the SLR allowed to identify the research lines where hybrid process artifacts have emerged, as well as the underlying motivations, use and characteristics. Moreover, the findings served to lay out a comprehensive research agenda about the issues requiring further research in the future. The outcome of this work, in turn, delivered a more profound characterization of hybrid process representations and a deepened understanding of the state of the art literature.

In the second contribution (`C2`), two studies were conducted to investigate the support of hybrid process artifacts during process modeling and model comprehension tasks. The former study examined the quality of the tool-support granted by hybrid artifacts during process modeling tasks. Therein, the findings suggested that hybrid process artifacts can provide convenient support to process modelers and can help to produce models with enhanced quality. Nevertheless, some shortcomings were identified in the tool-support. Overall, the outcome of this work revealed the use of hybrid process artifacts during modeling tasks, while the reported shortcomings served as a basis for an improved modeling tool-support (particularly within the DCR platform [LMMS19]).

The second study in contribution `C2` addressed the usage of hybrid process artifacts during model comprehension tasks. The analysis compared the way domain experts and IT specialists engage with hybrid process artifacts to complete tasks requiring the extractions of different types of information about the process. The results delineated the behavior of both user groups and showed how disparities in people's background and task type change the way process artifacts are perceived and used in practice. The analysis also provided deepened insights about the strategies underlying stakeholders' behavior when conducting comprehension tasks. The outcome of this work can be used to improve the support of existing process representations in order to better meet stakeholders' individual preferences and improve their understanding of declarative process models.

In the third contribution (`C3`), a catalog of modeling practices and a set of complexity metrics were proposed in two studies. The modeling practices, in the former study, originated from a set of quality dimensions inferring the implicit criteria used by experts to evaluate the quality of declarative process models in practice. These quality dimensions were elicited following a particular knowledge elicitation approach supported by PCT (Personal Construct Theory). The quality dimensions and the underlying modeling practices, identified as a result of this approach, were organized into a set of themes and categories addressing both the semantic and pragmatic qualities of declarative process models. Most importantly, the outcome of this work demonstrated the potential of PCT in conducting interpretive studies in the context of process modeling. In addition, it provided better understanding of the aspects defining the quality of declarative process models, which following the reported modeling practices can be taught to modelers to help them focus on the important quality dimensions and improve the modeling of business processes.

The second study in contribution `C3` proposed four complexity metrics (i.e., size, density, separability and relation variability). The factors captured by these metrics were evaluated in an empirical study and have been shown to affect the understandability of declarative process models. As a result, the outcome of this study delivered quantifiable means to assess the complexity of declarative process models and estimate the mental effort required to comprehend them. This contribution is crucial to ensure that declarative models can be easily communicated and interpreted by process stakeholders.

All in all, the framework presented in this thesis shall be seen as an important milestone towards a better modeling and comprehension of declarative process models. The studies conducted as part of this research explored many important aspects and reported several insights which will contribute to the development of more advanced modeling approaches and tool-support.

Future Work. There are many pertinent directions for future work, which can be followed into two main research streams: a stream delving into the comprehension of hybrid process artifacts, and another stream pursuing the research on quality dimensions and metrics for declarative process models. With regards to the former research stream, it is important to understand the way process stakeholders integrate information from different artifacts in their mind and investigate the mental effort associated with this cognitive process. Herein, it is possible that several factors (e.g., representation of information, the extent of information overlap between artifacts) influence the effort required to cognitively integrate the information incoming from different process artifacts, which would, in turn, impact their understandability and support for different tasks. While the carried-out research delineated the usage of hybrid process artifacts, more research is required to study the effects of different factors on the mental effort required to engage with hybrid process artifacts.

Regarding the second research stream, it is important to further investigate the quality dimensions that are unique to declarative languages (e.g., declarative modeling patterns shown in Table 5.1) and from there suggest new metrics capturing models' complexity with regards to these quality dimensions. Herein, it is possible that different types and combinations of constraints (i.e., relations in the DCR Graph) produce several effects and imply different kinds of dependencies in the model. These factors can potentially raise the complexity of declarative process models and thus induce increased levels of cognitive load. Understanding the impacts of these factors and capturing them in complexity metrics will complement the set of proposed metrics and therefore provide more accurate estimations of the overall complexity of declarative process models.

Part II

Articles

Chapter 7

Article 1: On the Declarative Paradigm in Hybrid Business Process Representations: A Conceptual Framework and a Systematic Literature Study

On the Declarative Paradigm in Hybrid Business Process Representations: A Conceptual Framework and a Systematic Literature Study

Amine Abbad Andaloussi[a,*], Andrea Burattin[a], Tijs Slaats[b], Ekkart Kindler[a], Barbara Weber[a,c]

[a] *Software and Process Engineering, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark*
[b] *Department of Computer Science, University of Copenhagen, 2100 København, Denmark*
[c] *Institute of Computer Science, University of St. Gallen, 9000 St. Gallen, Switzerland*

Abstract

Process modeling plays a central role in the development of today's process-aware information systems both on the management level (e.g., providing input for requirements elicitation and fostering communication) and on the enactment level (providing a blue-print for process execution and enabling simulation). The literature comprises a variety of process modeling approaches proposing different modeling languages (i.e., imperative and declarative languages) and different types of process artifact support (i.e., process models, textual process descriptions, and guided simulations). However, the use of an individual modeling language or a single type of process artifact is usually not enough to provide a clear and concise understanding of the process. To overcome this limitation, a set of so-called "hybrid" approaches combining languages and artifacts have been proposed, but no common grounds have been set to define and categorize them. This work aims at providing a fundamental understanding of these hybrid approaches by defining a unified terminology, providing a conceptual framework and proposing an overarching overview to identify and analyze them. Since no common terminology has been used in the literature, we combined existing concepts and ontologies to define a "Hybrid Business Process Representation" (HBPR). Afterward, we conducted a Systematic Literature Review (SLR) to identify and investigate the characteristics of HBPRs combining imperative and declarative languages or artifacts. The SLR resulted in 30 articles which were analyzed. The results indicate the presence of two distinct research lines and show common motivations driving the emergence of HBPRs, a limited maturity of existing approaches, and diverse application domains. Moreover, the results are synthesized into a taxonomy classifying different types of representations. Finally, the outcome of the study is used to provide a research agenda delineating the directions for future work.

Keywords:
Hybrid process model, Understandability of process models, Process flexibility, Declarative process modeling, Business process modeling

---

1. Introduction

In the development of today's Process-Aware Information Systems (PAIS), process modeling has become an important instrument to cope with the complexity of both the *management* and the *enactment* of business processes [1]. On the management level, process modeling provides input for requirements elicitation and allows concretizing business processes while ensuring a common understanding for both domain experts and IT specialists [2]. By deploying a variety of artifacts, process modeling provides a means for communication and collaborative design and enables benchmarking, optimization and process reengineering [3, 4]. The impact of process modeling goes beyond the management level to cover also the enactment level. Process modeling provides a blue-print for process execution, which in turn, facilitates system support and enables process enactment [5]. Furthermore, the outcome of process modeling enables a wide range of model analysis and verification techniques and allows simulating the model behavior under different execution scenarios [4].

The literature proposes a variety of approaches to graphically represent business processes as a process model. These approaches deploy different modeling languages (e.g., BPMN [6], Petri nets [7] , Declare [8], DCR [9]) and different types of process artifacts (e.g., process models, textual descriptions, animations and guided simulations). Depending on the kind of behavior implied in the process specifications, a business process can be most concisely described using a language from the *imperative–declarative* paradigm spectrum [10, 11]. Imperative languages allow describing explicitly the exact course of actions governing the execution of the business process which often makes them understandable to both domain experts and IT specialists. The use of imperative languages is suitable to model business processes where the execution alternatives are explicitly described in the process specifications. However,

---

[⋆] Work supported by the Innovation Fund Denmark project *EcoKnow* (7050-00034A).

[*] Corresponding author

Email addresses: `amab@dtu.dk` (Amine Abbad Andaloussi), `andbur@dtu.dk` (Andrea Burattin), `slaats@di.ku.dk` (Tijs Slaats), `ekki@dtu.dk` (Ekkart Kindler), `barbara.weber@unisg.ch` (Barbara Weber)

Preprint submitted to Journal of Information Systems *December 13, 2020*

some process specifications tend to abstract from describing the different execution alternatives and rather define a set of constraints guiding the overall process. These specifications can be naturally modeled using declarative languages, which allow using constraints to describe flexible business processes concisely. This way, it becomes possible to overcome the rigidity imposed by imperative languages and describe highly dynamic environments [12].

Previous research has provided evidence for the existence of business processes comprising both rigid and flexible parts [13]. Hence, restricting the modeling of business processes to declarative or imperative languages would imply an unnecessary complexity when modeling the rigid or the flexible parts of the business process. Since declarative languages take a constraint-based approach to describe the control-flow of business processes, representing the rigid parts using a declarative language, would require a high amount of constraints to impose a very specific behavior. Likewise, using an imperative language to model the flexible parts would require specifying all the possible execution alternatives, which would most likely result into a "spaghetti-like" model. (cf. Section 2.4.1 for a concrete example highlighting this inconvenience). In order to enable the modeling of both rigid and flexible parts of business processes concisely, a set of so-called "hybrid" approaches has emerged in the literature. While some approaches address the limitations of declarative notations and propose hybrid languages to combine declarative and imperative languages, other approaches address the separation of concerns between imperative processes and business rules and propose hybrid languages and hybrid process artifacts combining imperative process models with declarative artifacts.

The proposed hybrid approaches have not only the potential for providing concise process representations, but they can be also used to address the notorious limitations of declarative notations associated with their understandability and maintainability [14, 15, 16]. One of the key challenges in that regard is the inability of users to cope with process models with too many constraints [14]. Considering the rich and complex semantics of declarative languages (e.g., Declare) and all the possible ways in which constraints can interact, the understandability of declarative process models gets quickly hampered when dealing with complex processes [14]. The cognitive dimensions framework [17, 18] provides a reasonable explanation to that. Indeed, the interpretation of declarative process models is associated with an increased mental effort as the user is required to keep track of the states of all interrelated constraints while striving to interpret a declarative model. This task gets more complicated, when considering indirect constraints between activities (or so-called "hidden dependencies" [16]). Therefore, it is necessary to interpret the model as a whole rather than specific constraints in isolation. Given the limited capacity of humans' working memory [19] and the small amount of items a human memory can hold (i.e., 7 ± 2) [20], the interpretation of such models becomes very difficult. Hidden dependencies are also among the issues affecting the maintainability of declarative process models. Due to the complex entanglement of constraints, it becomes hard to determine which constraints are affected by a change of the specifications and to check the consistency of new changes with existing constraints [15]. Hence, the maintainability of declarative process models becomes easily prone to misalignment between the process specifications and the actual process model. To overcome the understandability and maintainability limitations of declarative languages, and to offer better support for the human cognitive processes associated with the modeling and the maintenance of declarative process models, several hybrid process artifacts supporting declarative artifacts with imperative ones have been proposed in the literature. These approaches address several issues associated with the understandability of declarative languages such as the complex semantics of declarative languages and the implications of hidden dependencies on the comprehension of declarative process models. Using hybrid process artifacts, extending declarative process models with imperative artifacts, the literature proposes several approaches to clarify the semantics of declarative process models and to track the implications of hidden dependencies on the interplay between the model activities [21, 22]. With regards to the maintainability of declarative process models, hybrid process artifacts can be used to address several challenges rising due to the continuous change of specifications. For instance, hybrid process artifacts (e.g., guided simulations supporting declarative process models) can be used to check the consistency of declarative processes after introducing new constraints in the model, and to keep track of the hidden dependencies rising from altering the constraints in the model [23].

In the following, we use the terminology "Hybrid Business Process Representation" (HBPR) to refer to *(1)* hybrid languages combining existing declarative and imperative languages and *(2)* hybrid process artifacts combining declarative and imperative artifacts.

1.1. Problem Statement

Hybrid approaches cover a wide range of representations addressing different aspects of process modeling. Although, these approaches share similar characteristics, the authors in the literature deploy a mix of terms to designate them, thus no common terminology exists. In addition, the literature lacks the basic foundations needed to define HBPRs. Besides a handful of publications (e.g., [24, 25]) describing HBPRs in an ad-hoc context, no framework allowing to structure and discern the characteristics of HBPRs has been proposed yet. As a result, the term *"hybrid"* becomes ambiguous and is sometimes used inconsistently in the literature. Furthermore, while several HBPRs have been surveyed in the context of supporting data intensive processes through data-centric approaches [26], little has been done to study the existing hybrid approaches taking a control-flow perspective to look into the declarative paradigm in hybrid representations. In the process of identifying the HBPRs proposed in this context, the need for a unified terminology and a conceptual framework providing a clear distinction of the different HBPRs proposed in the literature becomes a must.

1.2. Contributions

In this paper, we propose a conceptual framework for process artifacts, provide a unified terminology for HBPRs, perform a Systematic Literature Review (SLR) to investigate the existing HBPRs with a declarative language or artifact, and suggest an agenda for future research. Our contributions can be described as follows:

- C1: Propose a conceptual framework to discern the interactions between the different concepts defining a process artifact (cf. Section 2).
- C2: Instantiate the proposed conceptual framework to provide a unified terminology allowing to conceive the different types of HBPRs (cf. Section 3).
- C3: Perform an SLR to scrutinize HBPRs and organize them into a comprehensive taxonomy (cf. Sections 4 and 5). The study will cover hybrid languages combining declarative and imperative languages and hybrid process artifacts combining imperative process models with declarative process artifacts to present business processes concisely. Additionally, the study will focus on hybrid process artifacts extending declarative process models with imperative artifacts to overcome the challenges of declarative modeling languages.
- C4: Delineate a research agenda for future research (cf. Section 6).

Considering the lack of a unified terminology and a clear conceptual framework allowing to define HBPRs (cf. Section 1.1), Contributions C1 and C2 can be generalized to any type of HBPRs, while C3 and C4 focus on hybrid approaches taking a control-flow perspective to look into the declarative paradigm in hybrid representations.

1.3. Overview and Paper Structure

This section provides an overview on the different concepts discussed throughout this study. The aim is to familiarize the reader with the important notions and outline the structure of the paper. Section 2 discusses three important concepts i.e., *business process*, *language* and *process artifact*. A Business process is concretized as a process artifact using a language. A comprehensive definition of a business process and its underlying core *concepts* and *aspects* is presented in Section 2.1, while a set of relevant language characteristics (i.e., *syntax*, *semantics* and *language paradigm*) is defined and discussed in Section 2.2. These concepts provide the building blocks for a framework defining the general scope of a process artifact (cf. Section 2.3).

The proposed framework is instantiated in Section 3 to denominate the two types of HBPRs i.e., *hybrid languages* and *hybrid process artifacts*. As briefly outlined in the beginning of this section, a hybrid language allows expressing a process artifact using a combination of languages (usually from the imperative-declarative paradigm spectrum, cf. Section 2.2.2), whereas a hybrid process artifact allows concretizing a business process using more than one process artifact. This distinction provides a unified terminology (previously mixed in the literature) allowing to designate HBPRs consistently. Once established, a literature search is conducted following the research method presented in Section 4. The findings are scrutinized in Section 5, where the existing HBPRs are analyzed and compared on different levels. The results of the analysis come to support but also to enrich the proposed conceptual framework through a taxonomy discerning the characteristics of both hybrid languages and hybrid process artifacts. In Section 6, the main findings of this work are discussed and a research agenda is presented to guide the direction for the upcoming research. Last but not least, the threats to validity are discussed in Section 7, before concluding the paper in Section 8.

2. Conceptual Framework

This section introduces a conceptual framework defining the general scope for a process artifact. Following the existing terminology used within the BPM field, Sections 2.1 and 2.2 present the concepts associated with business processes and languages respectively. The interactions between these concepts are explained in Section 2.3, where the process artifact framework is presented. Finally, the different types of process artifacts are illustrated in Section 2.4.

2.1. Business Process

A *business process* is defined as *"a set of activities that are performed in coordination in an organizational and technical environment. These activities jointly realize a business goal."* [27]. The way a business process operates in the real world is captured by a process modeler as a set of *abstractions*, each emphasizing a given portion of reality. These abstractions are used to compose a subjective perception of the real world in the form of a mental model [28]. A *mental model* incorporates all the abstractions captured by the modeler about the way the business process operates in the real world. The shape of the mental model is affected by the *concepts* acquired by the modeler from the different ontologies proposed within the BPM field [28]. These concepts help the modeler to aggregate and structure the abstractions about the business process domain more efficiently [29]. Previous research [3] has classified these concepts into *(a) core concepts* which refer to the set of concepts defining the core elements of a business process i.e., process and tasks which instantiate into cases and activities [3] and *(b) aspects* which provide different lenses to look at the business process. The BPM literature discusses three main aspects: the *control aspect*, *organization aspect* and *information aspect*. In addition, several other aspects can be captured from a business process, for instance the *assignment aspect*, *security aspect* and *transaction aspect* [3].

This study focuses on the *control aspect* (also called *control-flow*), which is regarded as the most salient aspect in the literature [3]. *The control-flow represents information about the order of the activities or the constraints for their execution* [12].

2.2. Language

This section presents the characteristics of a language in terms of syntax and semantics (Section 2.2.1) and paradigm (Section 2.2.2).

2.2.1. Language Syntax and Semantics

A *language* is used to represent the business process. It is seen by Morris [30] as *"a system of interconnected signs, has a syntactical structure of such a sort that among its permissible sign combinations some can function as statements, and sign vehicles of such a sort that they can be common to a number of interpreters"*. A language can be a *natural language* (e.g., English) or an *artificial language* (e.g., programming language or conceptual modeling language). Among the features categorizing a language, Morris introduced *syntax* and *semantics*. Syntax is defined as *"the formal relation of signs to one another"* [30] i.e., the relations between finite meaningful elements of the language which allow deriving grammatically correct expressions. Semantics is defined as *"the relation of signs to real world entities they represent"* [30] i.e., the mapping between the language elements and the real world entities which allows conveying meaning [31].

Languages differ in terms of their *formality*, both regarding the syntax and the semantics. The syntax can be evaluated in terms of its grammatical structure and the completeness of its vocabulary, while the semantics can be evaluated in relation to the extent to which the semantic domain of the language is known [32]. Thereby, both syntax and semantics can be categorized as being *informal*, *semi-formal*, or *formal*.

2.2.2. Language Paradigm

The *language paradigm* can be seen as the style in which the language is written. Languages can be differentiated according to the *imperative–declarative* paradigm spectrum. This paradigm takes origins from the field of computer programming. As specified by Winograd [33], imperative programming is based on the idea that the *"the knowledge of a subject is intimately bound with the procedures for its use"*. In other words, imperative programming aims at specifying explicitly the set of commands leading to an output. Conversely, declarative programming as defined by Lloyd [34] aims at *"stating what is to be computed, but not necessarily how it is to be computed"*. Simply put, declarative programming aims at specifying the requirements to be achieved and letting the system determine the way to achieve them.

Roy and Haridi [35] use the notion of *state* to discriminate the two paradigms from a technical perspective. They defined a state as *"a sequence of values in time that contains the intermediate results of a desired computation"*. According to the authors a state can be either *implicit* or *explicit*. An implicit state is a state which neither the computational model nor the program are aware of, so it exists just in the mind of the programmer [35]. Conversely, an *explicit* state is a state which can be explicitly traced in the computational model and by the programmer. Following this distinction, declarative languages encode states *implicitly*, whereas imperative languages encode states *explicitly*. For instance, consider the two implementations of the factorial function shown in Figure 1. In the Prolog code (read as "the factorial of *Number* is *Factorial* if *Number*>0 and *Number1* is *Number*-1 and the factorial of *Number1* is *Factorial1* and *Factorial* is *Number*Factorial1*"), the notion of state is implicit. Indeed, although it would be possible for the programmer to trace down the sequence of states following a certain input value during the execution, there is no explicit predicate in the code keeping track of the computation result (i.e., state) after each recursive call. Alternatively, in the imperative Java implementation of the factorial function, the variable *value* is used in the computation model to explicitly keep track of the computation result after each iteration.

Listing (1) A program computing the factorial of *n* in Prolog

```
factorial(0,1).
factorial(Number, Factorial) :-
    Number > 0,
    Number1 is Number - 1,
    factorial(Number1,Factorial1),
    Factorial is Number * Factorial1.
```

Listing (2) A program computing the factorial of *n* in Java

```
public static int fact(int n)   {
    int value = 1;
    for(int i=n; i>0; i--)
        value *= i;
    return value ;
}
```

Figure 1: Declarative and imperative implementations of the factorial function in Prolog and Java.

The explicitness and the implicitness of states is bound to the representation layer of languages. This is because, at the execution layer, all languages are executed as a set of deterministic procedures which can be represented as states and transitions. This fact allows for a more fine-grained distinction between imperative and declarative languages. Hereby, we define imperative languages as *languages where states are explicit in both representation and execution layers*, and declarative languages as *languages where states are implicit in the representation layer and explicit in the execution layer*.

The same notion of *state* can be used when comparing imperative and declarative modeling languages. At the representation layer, imperative languages as defined by Pesic [14] allow modeling processes where *"all execution alternatives are explicitly specified"*, which in turn, enable representing the different states of a process explicitly. When modeled graphically, imperative languages provide a continuous trajectory (i.e., a sequence of states and transitions) allowing to reach any possible outcome allowed by the model [36]. The use of imperative languages implies the modeling of all possible courses of actions, which is only possible with a complete and well-detailed knowledge about all the alternative paths a business process execution might undergo. However, this is not always possible as the execution path of some business processes might depend on specifications that are only available at run-time and might also be unique to each process instance [37]. Alternatively, *declarative languages* as defined by Pesic [14] are language allowing to express models where *"constraints implicitly specify* [the] *execution alternatives as all alternatives that satisfy the constraints"*. In other words, declarative languages use constraints to describe the overall interplay of actions without explicitly describing the sequence of states and transitions leading to each particular outcome. Herein, states and transitions are implicit at the representation layer of the language and are only constructed at the execution layer, where the predicates and the formulas are interpreted [36]. This characteristic gives the capability to represent highly dynamic business processes concisely without having to explicitly specify the path of each single possible process execution.

Imperative and declarative languages allow representing different types of behaviors (i.e., forbidden, common and exceptional behaviors) [8], as shown in Figure 2, imperative languages are suitable to describe the common behavior, whereas declarative languages extend to both common and exceptional behaviors.

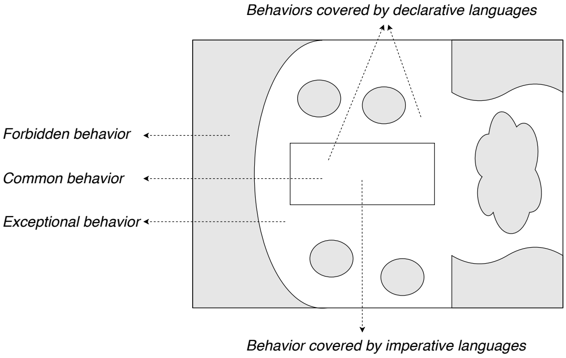


Figure 2: The different behaviors of a business process and the suitability of imperative and declarative languages to cover these behaviors. Adapted from [8].

Figure 3 depicts a three-dimensional framework describing the common languages used to model the control-flow aspect of business processes in terms of formality (both of syntax and semantics) and language paradigm. All these languages describe the control-flow as the order between the different process activities. In the figure, languages sharing more or less the same level of formality and language paradigm are grouped together. For example, Declare, DCR (Dynamic Conditional Response), CMMN (Case Management Model and Notation) [38] and XTT2 (Extended Tabular Tree version 2) [39] have a formal syntax, formal semantics and belong to the declarative language paradigm. Another group of languages comprises Petri nets, BPMN (Business Process Modeling Notation) and YAWL (Yet Another Workflow Language) [40] which all share a formal syntax, formal semantics and belong to the imperative language paradigm. Other languages with distinct level of formality and language paradigm are depicted individually. For instance, R2ML[1] (REWERSE Rule Markup Language) is characterized by a formal syntax, semi-formal semantics and belongs to the declarative language paradigm. Conversely, the original EPC (Event-driven Process Chains) language [41] has a formal syntax, informal semantics and belongs to the imperative language paradigm, whereas SBVR (Semantics of Business Vocabulary and Business Rules) [42] has a semi-formal syntax, semi-formal semantics and belongs to the declarative language paradigm.

Natural language in turn has an informal syntax and informal semantics. However, depending on the used grammatical structure and deployed language vocabulary, it can serve for expressing both declarative and imperative process specifications (cf. Figures 5a and 5b – the first part describes the interplay of actions using constraints, thus it is written in a declarative style, whereas the second part describes the explicit courses of actions, thus it is written in an imperative style). This is exactly why natural language is divided into *Imperative Natural Language* (I-NL) and *Declarative Natural Language* (D-NL). Although natural language has been used for a long time to describe business processes (e.g., paper-based documentations, regulatory documents), it has not been deployed in the BPM literature as a single artifact to represent a business process. However, with the emergence of hybrid process artifacts, natural language has been often combined with other imperative and declarative process representations. (e.g., [43, 44, 45, 46]).

2.3. Process Artifact

This section introduces the process artifact framework. Figure 4 combines the different pieces discussed in Sections 2.1 and 2.2 to illustrate our conceptual framework. The emergence of this framework can be seen as the result of putting together the existing concepts, which have been so far discussed in isolation in the literature. The relationships between the business process, the process artifact and the conceptualization entities are derived from the work of Axenath et al. [3]. The interactions between the process artifact, the mental model and the modeling concepts (represented within the conceptualization entity) are extracted from work of Soffer et al. [28] and the PhD thesis of Zugal [29]. Finally, the role of the language in bridging the gap between conceptualization and process artifacts is inspired by the ontological foundations proposed in Guizzardi's PhD thesis [32].

A process artifact is *the concretization of a business process using a language*. It is an *external* representation describing the way a business process operates in the real world in a formal or informal way [3]. It reflects the modeler's mental model which is an *internal* representation of the business process [28]. The core concepts and the different aspects (cf. Section 2.1) introduced within the BPM field constitute the conceptualization entity, which, in turn, allows structuring both the mental model

---

[1] See `http://www.macs.hw.ac.uk/bisel/rewerse/I1/oxygen.informatik.tu-cottbus.de/rewerse-i1%40q%3dr2ml.html`

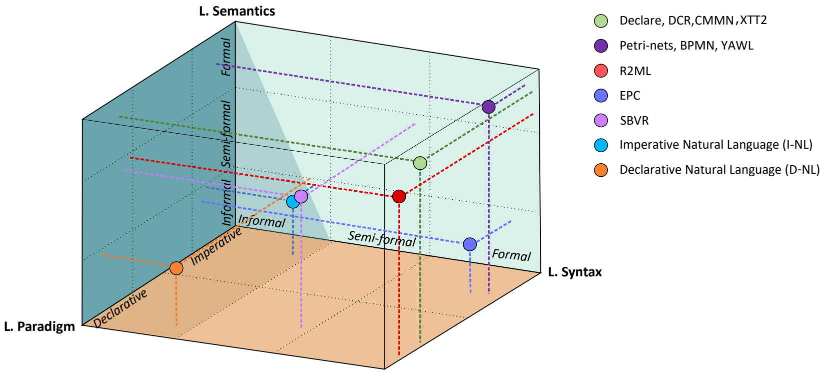


Figure 3: Categorization of some languages according to the formality of their syntax, the formality of their semantics and their language paradigm.

and the process artifact. The former is structured by providing a schema supporting the aggregation of knowledge about the business process more efficiently [29, 28]. The latter is structured through the concepts and ontologies governing the modeling of business processes [3]. The language has a central role within the process artifact framework. Indeed, by choosing an appropriate syntax, semantics and language paradigm, a language allows expressing a process artifact. In addition, the language enables expressing the different notions of the conceptualization entity [32], which offers a means to transfer the BPM knowledge and make it attainable to the modeler.

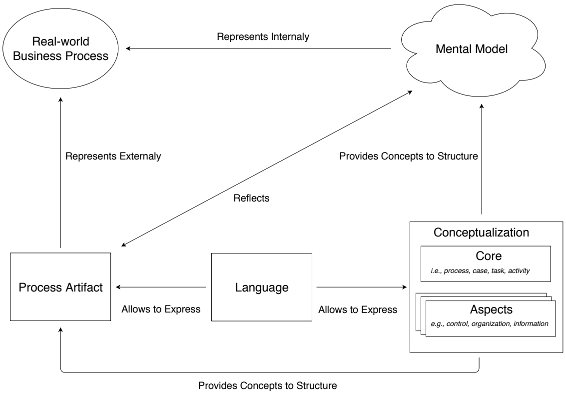


Figure 4: The process artifact framework.

The language provides the process artifact with a set of *inherent features* (i.e., syntax, semantics, paradigm, cf. Section 2.2). For instance, one can say that a process artifact is described in an imperative language characterized by a formal syntax and formal semantics (e.g., Petri nets). Besides the inherent features, a process artifact has a *visual feature*. Building upon the process visual representations in [47], a process artifact can be *static*, *dynamic*, or *interactive* (cf. Section 2.4 for examples). A static process artifact is characterized by a visual representation that remains static over time (i.e., textual process description, process model). A dynamic process artifact is characterized by an animated visual representation (i.e., dynamic over time) replaying previous executions of a business process (e.g., replay of event log traces). An interactive process artifact is characterized by an animated and interactive visual representation that changes depending on the way the user interacts with it (i.e., a guided simulation of a business process).

2.4. *Examples of Process Artifacts*

This section illustrates examples of the different process artifacts introduced in Section 2.3. Section 2.4.1 illustrates static artifacts. Section 2.4.2 illustrates dynamic artifacts. Section 2.4.3 illustrates interactive artifacts.

2.4.1. *Static Artifacts*

Textual process descriptions and process models are both examples of static artifacts, which means that their visual representation does not change over over time (compared to dynamic and interactive artifacts). Figure 5 shows two fragments describing the process of editing and handling a project proposal. Although both fragments are written in natural language, the interactions between the process activities are expressed differently. Indeed, while Fragment 1 describes the general interplay of actions in a loosely-coupled manner (using a D-NL), Fragment 2 specifies the exact course of actions with no room for flexible behavior (using an I-NL, e.g., *"Note that all decisions are final and cannot be reversed"*). This example illustrates a practical scenario where process specifications comprise both flexible and rigid requirements.

The two process models in Figure 6 describe the editing part of the project proposal process (cf. Fragment 1) using imperative (i.e, BPMN) and declarative (i.e, DCR[2]) languages respectively. Although, both models accommodate the same behavior, it is clear that the BPMN model in Figure 6a contains many more elements than the DCR model in Figure 6b, which results in a spaghetti-like process model, making it more visually-complex and thus, hard to understand and maintain. Hereby,

---

[2] The semantics of the DCR relations are summarized in `https://wiki.dcrgraphs.net/connection/`.

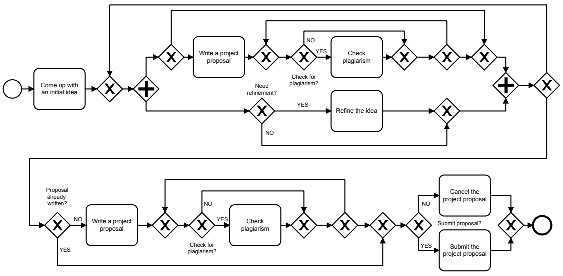
Fragment 1:
The process of writing a project proposal starts when the author comes up with an initial idea. Afterwards, it is possible to write a project proposal and to refine the idea at any time. After having written the project proposal it becomes possible to check for plagiarism. It is possible to cancel the proposal if it turns out that the idea is infeasible. Otherwise, as soon as the project proposal is described sufficiently well, it is possible for the author to submit the proposal. Note that a proposal can be submitted only once.

(a) Declarative process description in D-NL, adapted from [12, 48].

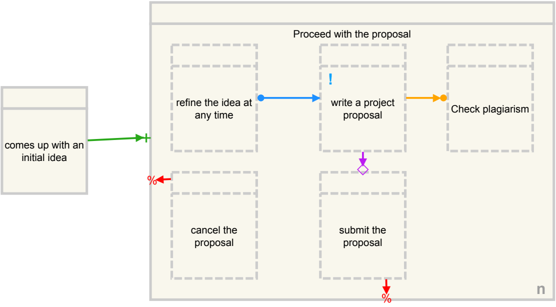
Fragment 2:
When a project proposal is received, a funding officer performs an initial screening of the proposal to check its compliance with the funding requirements of the institute. In case the proposal is not compliant, it is directly rejected. Otherwise, if the proposal complies with the given requirements, the funding officer provides an initial review of the proposal and sends it together with the initial proposal to a competent committee. The committee evaluates the proposal and based on their decision, the project proposal is either approved for funding or rejected. Note that all decisions are final and cannot be reversed.

(b) Imperative process description in I-NL.

Figure 5: Two process fragments describing different parts of the project proposal process.

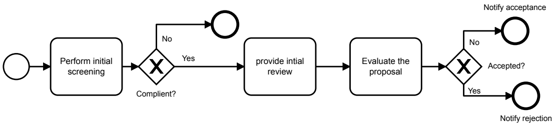


(a) Fragment 1 modeled imperatively.

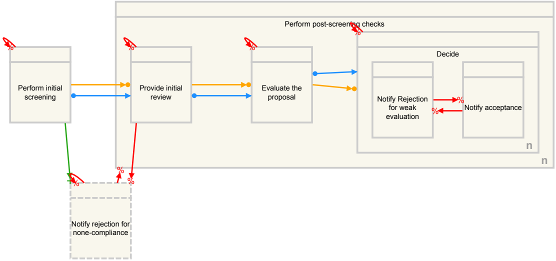


(b) Fragment 1 modeled declaratively.

Figure 6: Comparing the process in Fragment 1 when modeled imperatively using BPMN, and declaratively using DCR. The semantics of the DCR relations are summarized online^2.



(a) Fragment 2 modeled imperatively.



(b) Fragment 2 modeled declaratively.

Figure 7: Comparing the process in Fragment 2 when modeled imperatively using BPMN, and declaratively using DCR. The semantics of the DCR relations are summarized online^2.

it becomes evident that imperative languages are not the ideal candidates for modeling flexible processes. Similarly, the handling part of the project proposal process (cf. Fragment 2) is described using imperative and declarative languages in Figure 7. Since this part of the process is rather rigid, describing it using a declarative language (cf. Figure 7b) would imply extra constraints to restrict the process behavior, which in turn result in a visually-complex process model, making its understandability and maintainability difficult. Alternatively, the use of an imperative language can provide a more concise process model (cf. Figure 7a).

2.4.2. *Dynamic Artifacts*

Dynamic artifacts provide an animated visual representation allowing to perceive how an existing process instance evolve overtime. Figure 8 illustrates the frames of a Petri nets token animation replying a single execution trace. The trace contains the execution of a process model implementing Fragment 2. The Petri nets token animation allows replaying the executed process instances, which in turn provides a visualization of the actual executions of the business process [49]. The choice of Petri nets to illustrate dynamic artifacts is motivated by the concept of token replay [49]. Any other modeling language with a similar concept can be used to express a dynamic artifact.

2.4.3. *Interactive Artifacts*

Interactive process artifacts provide a dynamic visual representation allowing the user to test the different courses of actions allowed by the process model. The example depicted in Figure 9 illustrates a guided simulation of a business process. The process artifact depicted in the figure is an instance of DCR Graphs, however, the same guided simulation could be provided by instantiating any other language. This artifact allows performing a guided simulation based on the user input,

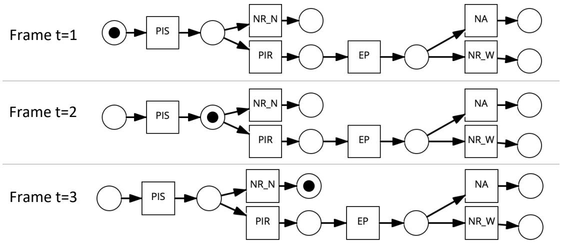
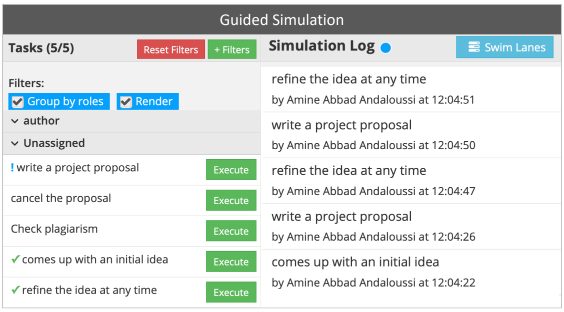


Figure 8: The frames of a Petri nets animation replaying a single event log trace. The letters in the transitions correspond to the initials of the activities extracted from Fragment 2 (e.g., NR_N: Notify rejection for none-compliance).



! write a project proposal
cancel the proposal
Check plagiarism
✓ comes up with an initial idea
✓ refine the idea at any time

refine the idea at any time
by Amine Abbad Andaloussi at 12:04:51
write a project proposal
by Amine Abbad Andaloussi at 12:04:50
refine the idea at any time
by Amine Abbad Andaloussi at 12:04:47
write a project proposal
by Amine Abbad Andaloussi at 12:04:26
comes up with an initial idea
by Amine Abbad Andaloussi at 12:04:22

Figure 9: Example of a guided simulation corresponding to Fragment 1.

which in turn allows to test the possible execution scenarios and to perceive the allowed behavior at any stage of the simulation.

3. Hybrid Business Process Representations

This section provides a unified terminology allowing to conceive the different types of HBPRs. As mentioned in Section 1, the definitions presented in this section are general enough to cover all HBPRs. Sections 3.1 and 3.2 instantiates the conceptual framework presented in Section 2 in order to define the scope of hybrid languages and hybrid process artifacts respectively and to highlight the properties allowing to denominate both of them. Finally, Section 3.3 provides a generic definition of a HBPR.

3.1. Hybrid Languages

A hybrid language combines existing languages at the level of their syntax, semantics and language paradigm. Considering the process artifact framework depicted in Figure 4, the language entity can be instantiated into a hybrid language which can be represented as any combination of languages from the imperative and declarative paradigm spectrum. The composition of a hybrid language is restrained by the ability to support the syntax, semantics and language paradigm allowed by all its composing languages. In other words, a hybrid language should remain consistent even when only the vocabulary of a single language is used [50]. This feature allows active users of a composing language to progressively adapt to the new hybrid language without having to acquire it from scratch [24].

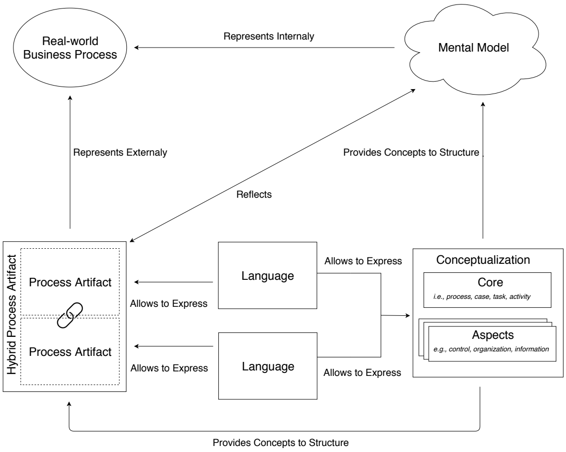


Figure 10: The scope of a hybrid process artifact defined based on the process artifact framework.

The use of hybrid languages brings a number of advantages. Firstly, by combining languages from the imperative–declarative paradigm, hybrid languages allow overcoming the limitations of individual languages and maintaining the balance between understandability and flexibility (e.g., [51]). Secondly, hybrid languages allow delivering an adequate language capable of representing business processes more concisely and precisely (e.g., [52]). Finally, hybrid languages enable the modeling of both rigid and flexible parts of a business process using the same language (e.g., [8]).

3.2. Hybrid Process Artifacts

A hybrid process artifact combines a set of interrelated process artifacts describing the same business process. Figure 10 illustrates an instance of the process artifact framework capturing the scope of a hybrid process artifact. Here, a set of languages (represented in the figure as separated entities) is used to describe different *but interrelated* process artifacts. The overlap between artifacts is an important characteristic discerning hybrid process artifacts from *multi-perspective* process models (e.g., [53]). While a multi-perspective process model describes each business process aspect in a detached artifact, a hybrid process artifact overlaps in describing the business process aspects, which in turn provides parallel visual representations where equivalent information can be extracted easily.

Hybrid process artifacts have been proposed in the literature to address the separation of concerns between imperative business processes and business rules. Moreover, hybrid process artifacts have been used to improve the understandability of process models as they provide hybrid visual representations allowing to clarify the semantics of the model and to extract equivalent information easily (e.g., [45, 21, 2]). Furthermore, hybrid process artifacts have the potential to improve the maintainability of declarative process models by providing concrete means to track the hidden dependencies (between the activities) introduced due to the entanglement of constraints in the model.

For instance, by extending a declarative process model with a guided simulation, it becomes possible to define the desired and prohibited behaviors (through test cases) and to constantly check them with the process model during a maintainability task (e.g., [23]). In addition, hybrid process artifacts can offer an alternative communication channels allowing to alternate between different levels of abstraction and support the knowledge transfer during the Process of Process Modeling (PPM) by combining formal and informal artifacts, which in turn, ensure a seamless communication between domain experts and IT specialists (e.g., [43, 54]).

3.3. Hybrid Business Process Representation

Following the characteristics of hybrid languages and hybrid process artifacts (cf. Sections 3.1 and 3.2 respectively), *a HBPR is defined as a collection of interrelated languages or process artifacts defining overlapping aspects and parts of the same business process.*

4. Literature Search Method

This SLR aims at identifying the existing HBPRs taking a control-flow perspective to look into the declarative paradigm. In addition, it focuses on providing a fundamental understanding of the proposed techniques through a transparent and reproducible approach. The research method deployed in this SLR follows the methodology proposed by Kitchenham [55] and lines up with the guidelines suggested by Budgen and Brereton [56] and Webster and Watson [57].

This section describes the search protocol adopted to conduct this SLR (cf. Figure 11). As a first step, the research problem is investigated by the authors and then formulated as a set of research questions (cf. Section 4.1). Then, following the recommendations of modeling experts (i.e., academics with several years of experience within the BPM field), a pilot search is conducted on the bibliography of notable authors within the field. (cf. Section 4.2). The outcome of this step allowed gathering the most common keywords and refining a comprehensive search string that covers the relevant literature (cf. Section 4.3). In addition, the most common publication venues are identified (cf. Section 4.4) and a set of inclusion and exclusion criteria is defined to filter the search results and select the most relevant articles (cf. Section 4.5). In the following step, the main literature search is performed and then the resulting articles are scrutinized (cf. Section 4.6). Afterwards a forward search and a backward search are performed (cf. Section 4.7). Finally, all the articles are read and relevant data are extracted according to a predefined scheme (cf. Section 4.8).

4.1. Research Questions

The research questions addressed in this study are the result of a series of meetings where the authors discussed the research problem (cf. Section 1.1) and the objectives of this study (cf. Section 1.2). In order to obtain a clear understanding about the HBPRs taking a control-flow perspective to look into the declarative paradigm, it is important to identify and investigate their distribution over time, type (i.e., journal, conference, book chapters) and venues. Therefore, the first research question is formulated as follows:
RQ1: *What publications about HBPRs taking a control-flow perspective to look into the declarative paradigm exist?*
This question is divided into the following sub questions:
RQ1.1: *How are these publications distributed over time?*
RQ1.2: *How are these publications distributed over publication type (i.e., journal, conference)?*
RQ1.3: *How are these publications distributed over publication venues?*

HBPRs have been deployed in several contexts. Therefore, it is possible that the proposed approaches have emerged within different research lines. The second research question investigates this aspect by addressing the following question:
RQ2: *What are the different research lines where the identified HBPRs were proposed?*

Once the research lines are identified, it is important to scrutinize the motivations behind the existing HBPRs in order to have a clear understanding about the process modeling issues which can be addressed using the proposed HBPRs. To this end, the third research question is formulated as follows:
RQ3: *What are the motivations driving the emergence of the identified HBPRs?*

The identification of the languages and the artifacts used to compose the existing HBPRs allows discerning the ones which were commonly deployed to model the imperative and the declarative process specifications. Hereafter, the fourth research question is formulated as follows:
RQ4: *Which languages and artifacts are combined in the identified HBPRs?*

Among the key contributions aimed by this work is a descriptive taxonomy allowing to categorize the existing HBPRs based on their common inherent and visual features. The fifth research question addresses this contribution as follows:
RQ5: *How can we categorize the identified HBPRs into a descriptive taxonomy?*

The maturity of the proposed hybrid approaches is another important aspect to investigate in order to evaluate the robustness of the proposed HBPRs. In that respect, it is necessary to investigate the extent to which the existing HBPRs have been formalized, and whether they have been implemented and evaluated empirically. These 3 aspects are addressed by the sixth research question as follows:
RQ6: *How mature are the identified HBPRs in terms of formalization, availability of implementation and empirical evaluations?*

Last but not least, the identification of the different application domains where HBPRs have been used allows illustrating the different fields where the use of HBPRs could be beneficial in practice. With this regard, the seventh research question is formulated as follows:
RQ7: *In what application domains can the deployment of the identified HBPRs be beneficial?*

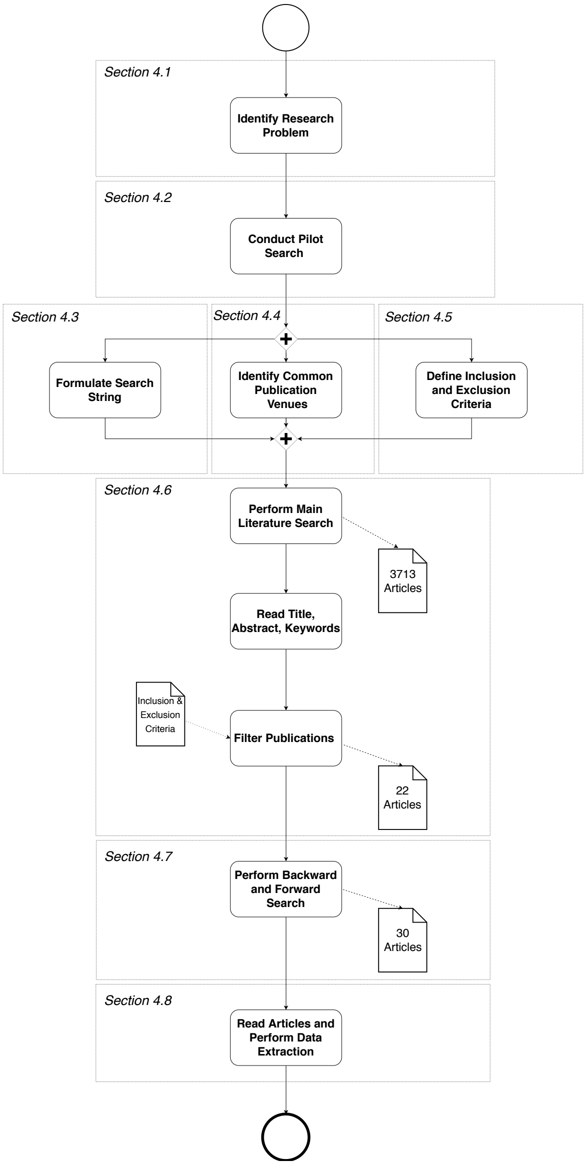


Figure 11: Summary of the protocol deployed to conduct this SLR (in BPMN language).

4.2. *Pilot Search*

Prior to the main literature search, a pilot search has been conducted on the bibliography of notable authors following the recommendations of modeling experts. As a result, the following articles were considered as reference: [46, 58, 50, 24, 16, 59, 21, 60]. These articles use different terminologies. Westergaard and Slaats [50] and Slaats et al. [24] use the terms, "*hybrid model*", "*hybrid process*" and "*mixing paradigms*" to describe HBPRs that combine languages from the imperative–declarative paradigm spectrum. Lu et al. [58] use the term "*flexible workflow*" in the context of an HBPR combining predefined parts with loosely coupled parts of a business process. Wang et al. [46] use the term "*integrated modeling*" referring to a HBPR that combines business rules with business process models. The authors in [16, 21, 59, 60] propose approaches that combine multiple artifacts. Although no clear hybrid terminology was mentioned, the proposed representations can be seen as HBPRs in a way that they combine interrelated artifacts. Therefore, it was necessary to extend the search to cover similar publications (incorporating a combination of declarative representations with other types of representations) where no hybrid terminology was mentioned.

4.3. Search String

Following the keywords extracted from the pilot search (cf. Section 4.2), the search string can be composed from the product of two sets of keywords: *(1)* keywords emphasizing the mixed nature of the proposed representations i.e., *hybrid*, *mixing*, *flexible* and *integrated*. *(2)* keywords emphasizing the concept of a business process i.e., *workflow*, *process*, *model* and *paradigm*. Additionally, as some articles do not use explicit terminologies to designate HBPRs, the keywords *declarative*, *constraint* and *rule-based* were added to Set *1*. These keywords allow covering all subsets of declarative representations including those extending declarative languages and artifacts with other types of representations (e.g., [16]). The use of these keywords leads to more false positive matches, but it helps covering a wider spectrum of the literature. In addition, the false positive matches are filtered-out using the inclusion and exclusion criteria introduced in Section 4.5 and the manual inspection of articles.

As some keywords might be used in different forms (e.g., appending suffixes), all keywords were transformed to their base form, then a wildcard character (i.e., asterisk *) was appended to each one of them to broaden the search by looking for all words starting with the same letters (e.g. *mix** → *mix, mixed, mixing*, etc.). Consequently, the following keywords were derived: *hybrid**, *mix**, *flexib**, *integrat**, *declar**, *constraint**, *rule**, *workflow**, *model**, *process** and *paradigm**. To interlink the search keywords, the *"OR"* logical operator was used. Indeed, during the study retrieval process, some literature search engines were unable to provide accurate results using complex search queries (i.e., search strings combining *"OR"*, *"AND"*, *"NOT"* operators). Thus, we have opted for a simple search string to provide a unified search string and maximize the hit rate across all search engines. The final search string is formulated as follows:

> hybrid* workflow* OR hybrid* model* OR hybrid* process* OR hybrid* paradigm* OR mix* workflow* OR mix* model* OR mix* process* OR mix* paradigm* OR flexib* workflow* OR flexib* model* OR flexib* process* OR flexib* paradigm* OR integrat* workflow* OR integrat* model* OR integrat* process* OR integrat* paradigm* OR declar* workflow* OR declar* model* OR declar* process* OR declar* paradigm* OR constraint* workflow* OR constraint* model* OR constraint* process* OR constraint* paradigm* OR rule* workflow* OR rule* model* OR rule* process* OR rule* paradigm*

4.4. Publication Venues

The notion of hybrid representations is widely deployed in several engineering fields. Therefore, conducting a general string look-up would lead to a huge amount of false positive matches. Once again, the recommendations of modeling experts were used to select the most popular data sources as well as the most prominent publication venues. Namely, the following data sources have been covered: Springer Link, IEE Explore Digital Library, ACM Digital Library, Science Direct, and Wiley Inter Science. As some data sources do not enable automated search, the Crossref[3] API was used. Within these data sources, the following journals have been covered: Decision support Systems (DSS), Information Systems (IS), Business & Information Systems Engineering (BISE), Software and Systems Modeling (SoSyM). In addition, the following conference venues have been considered: International Conference on Software and System Processes (ICSSP), Enterprise Distributed Object Computing (EDOC), Business Information Systems (BIS), Business Process Management (BPM), Business Process Modeling, Development and Support (BPMDS), Conference on Advanced information Systems Engineering (CAiSE), Conference on Conceptual Modeling (ER), Fundamental Approaches to Software Engineering (FASE), Formal Methods (FM), Integrated Formal Methods (IFM) and On the Move to Meaningful Internet Systems (OTM). Note that the proceedings of the forums and the workshops organized during each conference were also covered by the search.

4.5. Inclusion and Exclusion Criteria

In order to frame the search and to filter-out false positive matches, a set of inclusion and exclusion criteria has been defined.

4.5.1. Inclusion Criteria

A study is relevant if the following criteria apply:

- IC1: The study emphasizes the modeling of a HBPR.
- IC2: The study proposes a HBPR that includes at least one declarative language or artifact.
- IC3: The proposed HBPR focuses on the control-flow aspect (as defined in Section 2.1).

4.5.2. Exclusion Criteria

A study is excluded in case one of the following criteria apply:

- EC1: The study does not have the main focus on the modeling of HBPRs. (e.g., This excludes studies mining HBPRs i.e., [61, 62, 63].)
- EC2: The study is not published in English.

4.6. Main Literature Search and Selection Process

The main literature search yielded a considerable amount of matching articles. The look-up in the search engines covered the meta-data of the articles i.e., title, abstract and keywords.

---

[3] Crossref is Digital Object Identifier (DOI) registration agency indexing publications identified with a DOI from different data sources. See `http://crossref.org/`

In total 3713 articles were found. The high number of article retrieved from the search engines is typical for systematic literature reviews (e.g., [64]). In addition, since no unified terminology about HBPRs exists, the search string was formulated to be over-fitting for the purpose to cover a wide range of literature.

The selection process was performed by the corresponding author of this paper who followed a systematic approach to filter the literature articles based on a set of inclusion and exclusion criteria (cf. Section 4.5), which has been formulated and agreed by all the co-authors. Furthermore, in order to reduce any potential bias while selecting articles, the selection process has been constantly checked by the co-authors and borderline papers were discussed before deciding on their inclusion. Prior to the selection process, the meta-data of these articles were organized in a spreadsheet. During the selection process, the title of each retrieved article was scanned first in order to determine its relevance to the literature review (according to the inclusion and exclusion criteria). If the title is prominent, then, the abstract and keywords are inspected to further determine the article relevance. In case, the relevance remains doubtful, the article is fully read before being discussed internally by the co-authors. A spreadsheet with all the found studies and the decision on inclusion and exclusion is available online at `https://doi.org/10.5281/zenodo.3516661`.

As result, 22 relevant articles were selected from the main literature search. The selection also includes articles from the pilot search (cf. Section 4.2). The articles selected in the main literature search are the following: [46, 24, 52, 51, 45, 50, 59, 65, 58, 8, 37, 66, 25, 67, 68, 69, 70, 22, 16, 21, 60, 71] (the articles meta-data are presented after adding the backward search and forward search results, cf. Table 1). In the next step, a backward and forward search are conducted on these articles to gather additional relevant studies.

4.7. Backward and Forward Searches

To cover a wider range of the relevant literature, the initial search process has been expanded with a backward search and forward search. Section 4.7.1 shows the results of the backward search, Section 4.7.2 shows the results of the forward search.

4.7.1. Backward Search

The backward search examines the references cited in the literature to learn more about the foundation of the knowledge in question. In the context of our SLR, the backward search has covered the publications selected from the main literature search. During this process, we came across 4 new publications addressing our research subject. The following publications were then appended to the results of the main literature search: [72, 73, 74, 75].

4.7.2. Forward Search

The forward search examines new articles citing the literature and provides a follow-up on the development of the knowledge in question. In the context of our SLR, Google Scholar[4] was used to conduct the forward search on the publications selected from the main literature search. As a result, 4 new publications were retrieved and appended to the results of the main literature search: [76, 77, 78, 79].

4.8. Data Extraction

This section describes the data extraction process and lists the attributes used to answer each of the research questions introduced in Section 4.1. The data extraction process consists of extracting information relevant to the SLR according to a predefined data extraction scheme. The attributes depicted in Figure 12 summarize the scheme used to organize the data.

The extended article meta-data, including the title, authors, keywords, abstract, references, publication year, publication type (i.e., journal or conference) and publication venue, are used to answer RQ1 by first enabling the identification of existing literature and then describing their time distribution, types, and venues. For this analysis, descriptive statistics (i.e., count and distributions in percentage) are used (cf. Section 5.1). To Answer RQ2, all the articles are read and labeled subjectively according to their research line. At the analysis, the results from the backward search are used to investigate the assigned labels and validate the identified research lines. During this process, a graph visualization depicting the cross-referencing between the articles in the literature is generated (cf. Section 5.2.1). For RQ3, the motivations behind each proposed approach are extracted, then used in the analysis phase to discern the different motivations driving the emergence of new HBPRs and to organize them into different categories (cf. Section 5.3). RQ4 is answered by looking at the combinations of languages and artifacts seen in the literature. In this regard, the artifacts used to construct each HBPR (together with their languages) are extracted, then both their inherent and visual features are scrutinized in order to discern their hybrid properties. (cf. Section 5.4). The data extracted in RQ4 is further investigated to answer RQ5 (cf. Section 5.5). To answer RQ6, data about formalization, availability of implementation and availability of evaluation are inferred. On that matter, the articles are classified as having a mathematical formalization, a meta-model formalization or not being formalized at all. Concerning the implementation, in case an article provides an implementation, information about tool name, type (i.e., prototype, plugin or commercial product), parent framework and reference are extracted. Then, for the evaluation, information about the number (#) of participants, evaluation reference, evaluation type (i.e., quantitative, qualitative or both), research aspects, instruments, measurements, and outcome are extracted. The data about formalization, implementation and evaluation are investigated during the analysis to denote the maturity of the proposed approaches (cf. Section 5.6). Finally, to answer RQ7, information about the different application domains used to exemplify the approaches proposed in the literature are extracted, grouped and presented at the analysis (cf. Section 5.7).

---

[4] See `https://scholar.google.com`

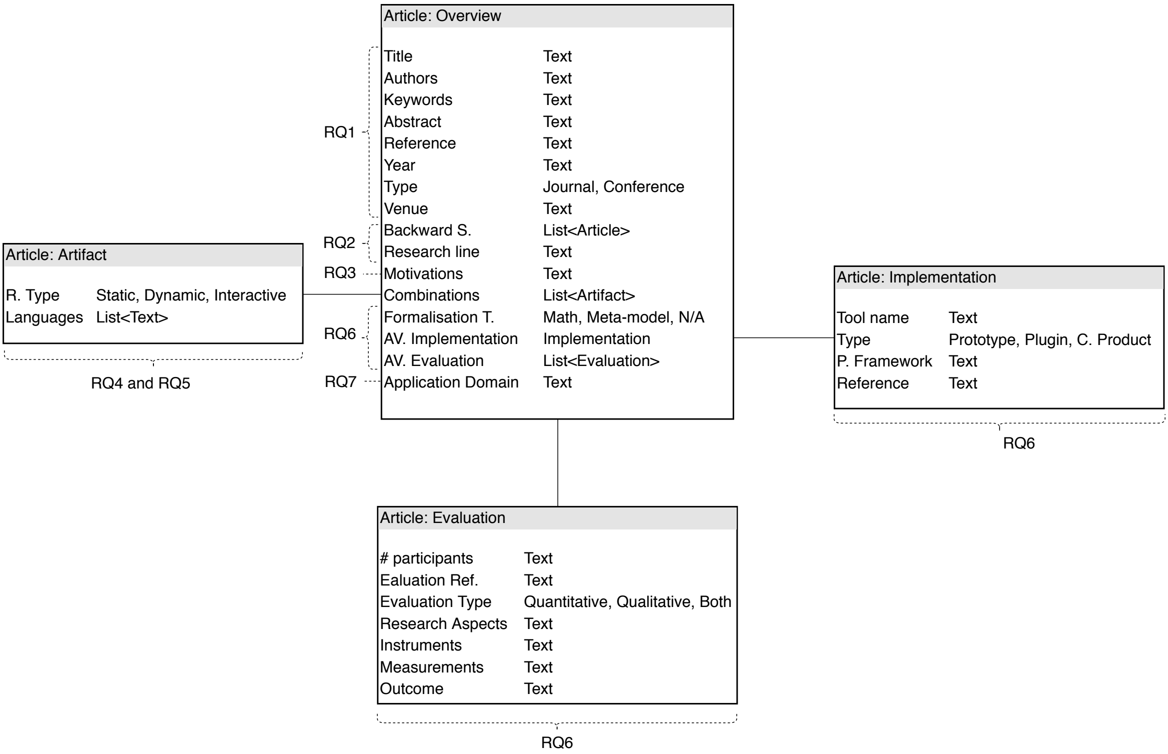


Figure 12: Data extraction scheme.

5. Analysis of Findings

This section provides an overarching analysis of the literature. Section 5.1 identifies the existing HBPRs and provides descriptive statistics (i.e., time distribution of publications, publication venues, and publication types) emphasizing the general findings obtained from the SLR search. Section 5.2 identifies the existing research lines that propose HBPRs. Section 5.3 highlights the motivations supporting the proposed representations. Section 5.4 distinguishes the different combinations of languages and artifacts proposed in the literature, Section 5.5 introduces a new taxonomy to categorize existing HBPRs, Section 5.6 investigates their maturity in terms of formalization, availability of implementation and empirical evaluation, and finally Section 5.7 reports the different application domains where the proposed HBPRs can be deployed.

5.1. Literature Search Findings

This section reports the results for RQ1. Table 1 shows the final study retrieval list including the title, the venue, the year and the authors of the selected articles. Overall, 30 articles were identified following the search method introduced in Section 4.

As depicted in Figure 13a, the articles addressing HBPRs have emerged since 2001 (answering RQ1.1). The time distribution of publications shows that 2011 and 2016 were the years with the highest number of publications addressing the topic (4 articles each year). By comparing the time distribution between the last two decades (cf. Figure 13b), one can notice that 77% of the articles were published between 2009 and 2018 compared to only 23% between 2001 and 2008. This shows an increasing tendency of articles proposing HBPRs over the last two decades. This tendency is also visible from the trend line depicted in Figure 13a.

Figure 13c shows the distribution of publications over publication type (answering RQ1.2). The search results show that 73% of articles appeared in conference proceedings, 23% appeared in journal proceedings, whereas only 4% where published as book chapters.

Finally, the selected articles were published in different venues (answering RQ1.3). As shown in Figure 13d, besides the initial publication venues considered for the main search, new venues have been covered during the backward and forward search. The distribution of the publication venues shows that EDOC, BPM and CAiSE take the lead with the largest proportions (i.e., 20%, 17%, and 10% respectively) and gather 47% of publications.

5.2. Research Lines

This section discusses the different research lines where HBPRs were introduced (answering RQ2). In the process of examining the articles retained in the final study retrieval list

Titles	**Venues**	**Years**	**Authors**
The Process Highlighter: From Texts to Declarative Processes and Back [21]	BPM	2018	Hugo A. Lopez et al.
Formal Model of Business Processes Integrated with Business Rules [79]	ISF	2018	Kluza, Krzysztof and Nalepa, Grzegorz J.
Discovering hidden dependencies in constraint-based declarative process models for improving understandability [45]	IS	2018	De Smedt, Johannes et al.
Effect of Linked Rules on Business Process Model Understanding [46]	BPM	2017	Wang, Wei et al.
The Semantics of Hybrid Process Models [24]	OTM	2016	Slaats, Tijs et al.
Improving Understandability of Declarative Process Models by Revealing Hidden Dependencies [67]	CAiSE	2016	De Smedt, Johannes et al.
Web-Based Modelling and Collaborative Simulation of Declarative Processes [70]	BPM	2016	Marquard, Morten et al.
Business Process Flexibility and Decision-Aware Modeling—The Knowledge Work Designer [60]	Book	2016	Hinkelmann, Knut
Mixed-paradigm process modeling with intertwined state spaces [52]	BISE	2015	De Smedt, Johannes et al.
Declarative Process Modeling in BPMN [51]	CAiSE	2015	De Giacomo, Giuseppe et al.
Hybrid Process Technologies in the Financial Sector [22]	BPM	2015	Debois, Søren et al.
Mixing Paradigms for More Comprehensible Models [50]	BPM	2013	Westergaard, Michael and Slaats, Tijs
Towards the Combination of BPMN Process Models with SBVR Business Vocabularies and Rules [78]	ICIST	2013	Mickevičiūtė, Eglė and Butleris, Rimantas
Creating Declarative Process Models Using Test Driven Modeling Suite [16]	CAiSE	2012	Zugal, Stefan et al.
Enriching Business Processes with Rules Using the Oryx BPMN Editor [73]	ICAISC	2012	Kluza, Krzysztof et al.
Patterns for Flexible BPMN Workflows [59]	EuroPLoP	2011	Zimmermann, Brigit and Doehring, Markus
Modeling Flexible Business Processes with Business Rule Patterns [65]	EDOC	2011	Milanovic, Milan et al.
Framework for Business Process and Rule Integration: A Case of BPMN and SBVR [68]	BIS	2011	Cheng, Ran et al.
Toward enhanced life-cycle support for declarative processes [69]	SEP	2011	Zugal, Stefan et al.
Exploiting Rules and Processes for Increasing Flexibility in Service Composition [72]	EDOC	2010	Sapkota, Brahmananda and van Sinderen, Marten
Flexibility as a Service [75]	DASFAA	2009	van der Aalst, W. M. P. et al.
On managing business processes variants [58]	DKE	2009	Lu, Ruopeng et al.
Towards a Language for Rule-Enhanced Business Process Modeling [74]	EDOC	2009	Milanovic, Milan and Gasevic, Dragan
Achieving Business Process Flexibility with Business Rules [66]	EDOC	2008	van Eijndhoven, Tim et al.
DECLARE: Full Support for Loosely-Structured Processes [8]	EDOC	2007	Pesic, Maja et al.
Patterns of Business Rules to Enable Agile Business Processes [71]	EDOC	2007	Graml, Tobias et al.
Specification and validation of process constraints for flexible workflows [25]	IS	2005	Sadiq, Shazia W. et al.
A constraint specification approach to building flexible workflows [77]	RPIT	2003	Mangan, Peter and Sadiq, Shazia
On Building Workflow Models for Flexible Processes [76]	ADC	2002	Mangan, Peter and Sadiq, Shazia
Pockets of Flexibility in Workflow Specification [37]	ER	2001	Sadiq, Shazia et al.

Table 1: Final study retrieval list. The publication venues are abbreviated as mentioned in Section 4.4. The new venues are the following: DASFAA (International Conference on Database Systems for Advanced Applications), DKE (Data and Knowledge Engineering), IS (Information Systems), SEP (Software: Evolution and Process), ICAISC (International Conference on Artificial Intelligence and Soft Computing), ISF (Information Systems Frontiers), Australasian Database Conference (ADC), EuroPLoP (European Conference on Pattern Languages of Programs) and RPIT (Research and Practice in Information Technology).

(cf. Table 1), a label was subjectively assigned to each article depending on its corresponding research line (cf. Section 4.8). The results of the labeling indicate that HBPRs have evolved in different contexts. More specifically, we have identified two main research lines within the BPM field. In the first research line (RL1), the authors proposed extending and supporting declarative languages, whereas in the second research line (RL2), the authors proposed integrating business rules with business processes. Section 5.2.1 validates the subjective labeling used to identify the research lines, and then Sections 5.2.2 and 5.2.3 present respectively the articles identified within these research lines.

5.2.1. Labeling Validation

To substantiate our subjective labeling, a backward search was performed on the articles of the final study retrieval list (cf. Table 1). During this process, an automated tool[5], was used to extract and filter the the references cited by each article in order to retain only citations referring to articles from the final study retrieval list. Afterwards using a graph visualization, articles were represented as nodes, and references were represented as directed edges between referring and referred articles. The size of each node has been defined based on the number of incoming edges (i.e., the number of cites by the other articles in the graph). As we suspected that the articles belong to different research lines, a color was assigned to each node based on the label of its research line . The resulting graph is depicted in Figure 14.

Accordingly, two nearly independent clusters of articles can be discerned. Namely, the cluster of articles in RL1 (colored in red) and the cluster of articles in RL2 (colored in green). Except a unique edge (cross-reference) between van Eijndhoven et al. [66] and Sadiq et al. [37], none of the other articles in one cluster has cited articles from the other cluster. By closely inspecting the context where [66] cited [37], we have found a single citation that came up in the context of sharing the same concerns as Sadiq et alẇith regards to process flexibility. However the two approaches use different concepts and techniques to define HBPRs.

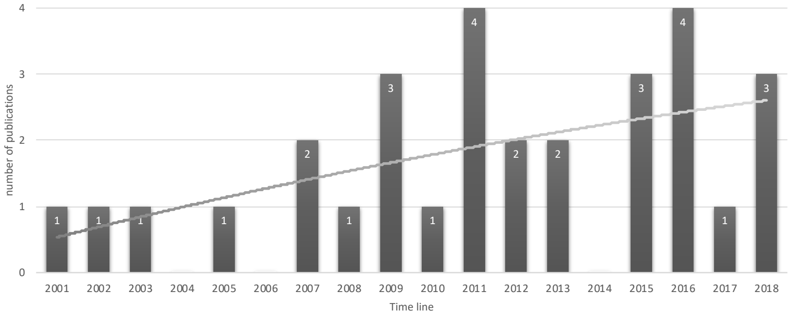
Additionally, one article (i.e., Hinkelmann [60]) remained independent from both research lines. Although, the approach proposes a hybrid representation addressing the separation of concerns between process and business logic, no cross-referencing has been identified in relation with the articles of RL2. By scrutinizing the article, we have noticed that the author did not explicitly describe the related work about the similar hybrid representations, which could in turn explain the lack of connections with other similar articles.

The size of the nodes provides insights about the approaches which offer the most widespread traction and were used as a basis to develop other approaches. In the context of the HBPRs covered in this study, [8], [37] and [50] were the most cited approaches in RL1, while [71] was the most cited approache in RL2. The impact of these approaches is discussed in Section 6.

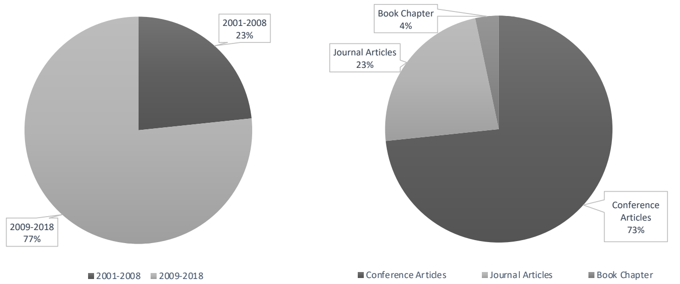
In the next sub-sections (cf. Sections 5.2.2 and 5.2.3), the articles proposed within each research line are introduced.

---

[5] CERMINE is the tool used to extract references from the articles. Source code available at `https://github.com/CeON/CERMINE`

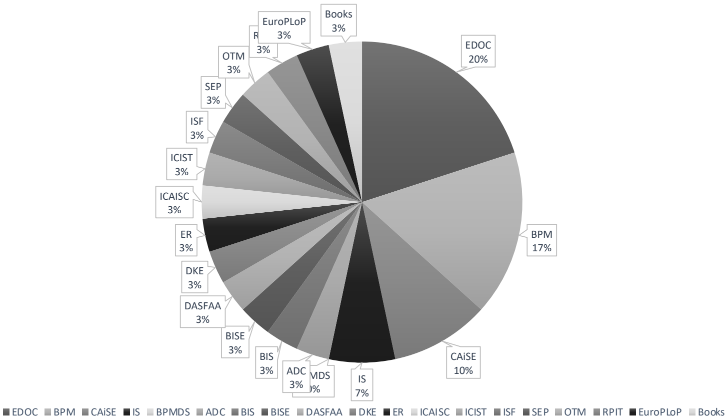


(a) Time distribution of articles organized by their year of publication.



(b) Time distribution of articles in the two last decades.

(c) Distribution of the articles based on their publication venue type.



(d) Distribution of the articles based on their publication venue.

Figure 13: Different distributions of the articles in Table 1.

5.2.2. *Extending and Supporting Declarative Languages*

The use of HBPRs extending declarative languages occupies a large part of the literature. Sadiq et al. [37] propose a hybrid approach to construct process models. Herein, an imperative process model is extended with several “pockets of flexibility” where the relations between the activities in each pocket are modeled in a declarative – constraint-based style. Pockets of flexibility are defined at design-time and explicit workflow executions are constructed at run-time according to the pockets’ constraints. Using the same concept, Mangan and Sadiq [76, 77] extend this work and propose a HBPR combining basic imperative modeling constructs with dynamic constraints (cf. Section 5.4). The later is further elaborated in [25]. Similarly, Lu et al. [58] present a HBPR combining pre-defined model parts with loosely coupled model parts using a constraint-based approach.

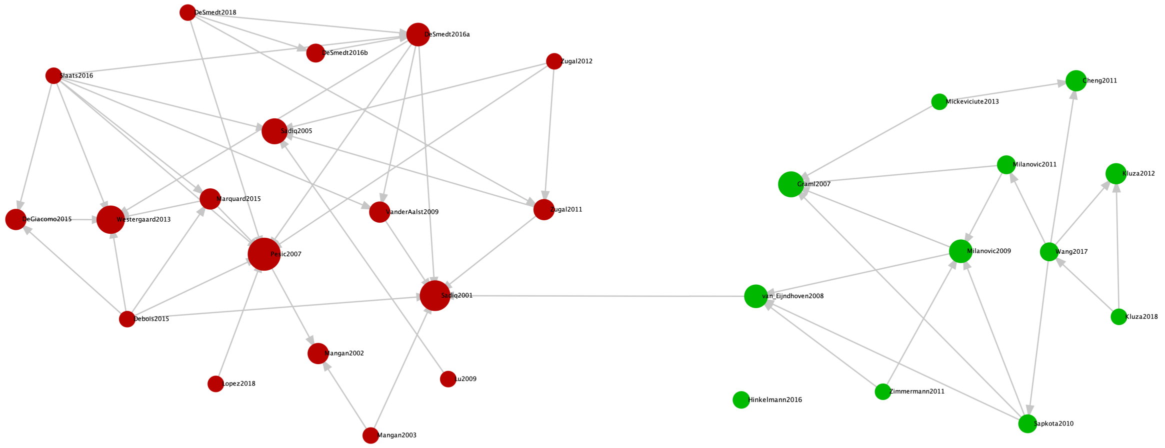


Figure 14: Cross-referencing graph of the articles in Table 1. The colors of the nodes refer to their corresponding research lines. The direction of the arrows indicates the citing between articles.

Pesic et al. [8] introduce *Declare* as a new constraint-based process modeling language for loosely coupled process models. Since Declare is not appropriate for modeling highly structured processes due to its constraint-based nature and the inability of users to make sense of process models with too many constraints [14], the authors suggested to combine it with YAWL (i.e., an imperative language). This in turn, would enable modeling both highly structured parts and loosely structured parts of process models efficiently. An extension for this approach was later proposed by Van der Aalst et al. [75].

Westergaard and Slaats [50] proposed a hybrid language allowing for the use of both the imperative and declarative formalisms within the same process model. This in turn, enables a concise representation for unstructured processes and detailed specifications for structured processes. Similarly, De Giacomo et al. [51] propose a new extension of BPMN entitled BPMN-D. With the same concept in mind, Slaats et al. [24] propose a hybrid framework, allowing to model processes in a hierarchical structure such that each sub-process of the hierarchy can be modeled using either an imperative or declarative language.

De Smedt et al. [52] address the trade-off between understandability and flexibility when using an imperative or a declarative modeling language, and analyze the implications of combining languages from different paradigms (particularly in terms of their syntax, execution semantics, and understandability). On these grounds, the authors introduce a step-wise approach to derive a hybrid language.

Zugal et al. [69, 16] propose a Test Driven Modeling (TDM) approach addressing the understandability and maintainability of declarative process models. The authors base their approach on the concept of computational offloading from CLT [80], which refers to the extent to which a user can extract certain information from a business process model quickly. As declarative process modeling languages have lower computational offloading (compared to imperative languages), enriching them with test cases would increase their computational offloading by providing means to easily locate hidden dependencies and validating specific scenarios against forbidden behaviors. Similarly, Marquad et al. [70] and Debois et al. [22] propose HBPRs to support declarative process models using guided simulations.

Lopez et al. [21] introduce the process highlighter as a means to clarify the semantics of declarative process models. The proposed hybrid representation interlinks the modeling constructs (i.e., roles, activities and constraints) of the model with the corresponding fragments in the textual description. This hybrid approach, in turn, provides a better alignment between the two artifacts and improves the comprehension of the model [48].

In another quest to support declarative process models, De Smedt et al. [67, 45] address the issue of hidden dependencies and its negative impact on the understandability of declare process models (cf. Section 1). In this vein, a hybrid representation revealing hidden dependencies is proposed to avoid all sort of ambiguities while conjoining Declare constraints.

5.2.3. *Integrating Business Rules with Business Processes*

The second research line evoking HBPRs in the literature is associated with approaches addressing the separation of concerns between imperative processes and business rules. These approaches aim at integrating business rules with process models. In this context, business rules are extracted from the control flow and represented in natural language or following specific rule-based languages (usually using an intuitive syntax). This way, the complexity of the process model is reduced and higher flexibility and adaptability are ensured at run-time. What makes the difference between the approaches proposed in this research line compared to the previous one is that, here, constraints are represented in a more human readable way (i.e., annotations), which in turn, improve the understandability of the business process. Business rules can be appended to a process model as model annotations [68] or as linked rules connected to specific parts of the process model [72]. In both cases, business rules provide a HBPR combining two different artifacts (i.e., business rules with a process model).

The literature proposes a variety of approaches aiming at combining business rules with process models. Cheng et al. [68] address four important aspects to consider while combining both artifacts: *(1)* Business rules and imperative process models have different representations. In other words, business rules tend to be represented textually while imperative process models tend to be represented graphically. Thus, the combination should map the textual annotations to the business process constructs with a minimal information loss. *(2)* The semantics of both artifacts are fundamentally different. *(3)* Each of the artifacts targets different levels of abstraction. While imperative process models elicit the way activities should happen, business rules emphasizes what should happen. *(4)* Nevertheless, an overlap between the specifications provided by both artifacts is possible which should be also taken into consideration when using such combinations. The authors propose an overarching framework to support the combination of imperative models and business rules by introducing two mapping methods allowing to identify the inconsistencies between business rules and imperative business processes. Similarly, Mickeviciute and Butleris [78] investigate the combination capabilities of imperative and declarative languages and address a possible mapping of elements of both languages to infer inconsistencies and overlaps.

Kluza et al. [73] claim that imperative languages are not suitable to model low-level logic of tasks in business processes. Alternatively, the authors propose to use business rules to integrate the low-level logic in business processes. In this work, the authors address two important integration aspects: *(1)* The visual modeling aspect of business rules by introducing an approach to manage them visually. *(2)* The execution aspect of the integrated model by providing an execution environment implementing a rule engine. Kluza and Nalepa [79] extend this work by introducing a formal semantics for integrating business rules in a process model.

Graml et al. [71] argue that existing business processes (modeled in an imperative style) lack adaptability and are unable to cope with changes in real-time. To ensure a high flexibility and better adaptability, the authors propose a set of modeling patterns allowing to extract derivation rules (used for decisions), constraints (used to enforce decisions), and process rules (used to define the dependencies between the process activities) from process models. These rules can be defined separately from the control flow and then integrated as linked rules. Following this approach, the resulting HBPR combines a process model and business rules modeled as linked rules. Similarly van Eijndhoven et al. [66], propose a rule-based approach to separate business rules from process models. The approach consists of of first discerning the static and the changing parts in the process model, then representing the latter as business rules. This way, all modifications on the changing part of the process will involve only editing the business rules.

Milanovic and Gasevic [74] introduce a HBPR allowing to model the different modeling patterns proposed in [71] to integrate business rules in business processes. This work was further elaborated in [65] by Milanovic et al., where the authors review the modeling patterns proposed in [71] and address the lack of a systematic modeling approach that abstracts from the implementation details and rather focus on the modeling itself. Consequently, the authors refined the approach proposed in [71] and abstracted it from any technology dependency.

Similarly, Zimmermann and Doehring [59] introduce a HBPR aiming at extracting the contextual facets from the process model and representing them as business rules. Hence, reducing the complexity of the process model. This way also, dynamic changes are supported and process instances are able to adapt to events and changes of context variables at run-time.

Sapkota and van Sindern [72] address the continuous change in business demands and the inability of existing service composition techniques to cope with flexibility and adaptability of business processes. Consequently, the authors consolidate between declarative and imperative designs by deploying business rules to define constraints and handle service orchestration in a dynamic manner. Similarly to Graml et al. [71], the authors emphasize the importance of extracting rules from process models, then integrating them in a way that business processes can adapt to changing requirements and ensure rules consistency without altering the composition logic.

The understandability of HBPRs combining a process model with business rules was investigated by Wang et al. [46]. In the design of their experiment the authors deploy a layout where a process model and linked rules are displayed side by side. This layout illustrates a hybrid representation that combines a process model with business rules expressed in natural language.

Hinkelmann [60] proposes a hybrid representation allowing to represent process logic imperatively and business logic declaratively in an integrated manner. As mentioned in Section 5.2.1, although no direct connections can be established (in term of cross-referencing) with the article of RL2, the proposed approach is clearly addressing the separation of concerns between process logic and business logic, and thus sharing similar characteristics with the other approaches in RL2.

5.3. Motivations

This section presents the motivations driving the emergence of the proposed HBPRs (answering RQ3). In the context of this study, the motivations are derived based on the functional aspects of the proposed approaches. This information is directly extracted from each of the selected articles after being fully read.

The articles published within the different research lines share several motivations. Most of the authors in the literature motivate their approaches and explicitly describe the motivations behind the proposed HBPRs. Through the literature, the following motivations were identified:

(a) Enhancing process flexibility and allowing adaptability at run-time by combining loosely structured model parts with highly structured parts.

(b) Reducing the complexity of process models by separating business rules from the control flow, then integrating them in a hybrid representation.

(c) Introducing hybrid languages to deliver the most adequate language allowing to represent business processes more concisely and precisely.

(d) Improving the understandability of process models and fostering the communication between the different process stakeholders (i.e., domain experts and IT specialists).

(e) Improving the maintainability of process models and ensuring better process re-usability.

(f) Supporting the modelers in the PPM.

Table 2 summarizes the motivations driving the emergence of the approaches proposed within RL1 and RL2. The table shows that several articles intersect with more than one motivation despite their context or research line. By looking at the distribution of the motivations over the two research lines, one can notice that both research lines share motivations about process flexibility, understandability and maintainability. However, only approaches in RL1 aim at improving the PPM and enhancing the conciseness and preciseness of hybrid languages, whereas only approaches in RL2 aim at separating and integrating business rules.

5.4. Combined Languages and Artifacts

This section identifies the languages and artifacts combined in the proposed HBPRs (answering RQ4). As shown in Table 3, some languages were commonly deployed in several hybrid approaches to model either imperative or declarative process specifications. With respect to RL1, Declare was the most common language to be combined with other languages (i.e., [75, 8, 50, 24, 52, 51, 67, 45, 69, 16]). Besides Declare, DCR was combined with textual annotations (i.e., [21]) and language independent representations (i.e., flow-based representation [22] and guided simulation tool [70]) in order to model declarative specifications. The proposed HBPRs were represented using different types of artifacts. On that matter, two distinct types of artifacts can be discerned, namely, static artifacts and interactive artifacts. Static artifacts were represented as a process model (i.e., [75, 8, 50, 24, 52, 51, 67, 45, 69, 16, 70, 22]) and a textual description (i.e., [67, 45, 21]). Whereas interactive artifacts were represented as a guided simulation (i.e., [69, 16, 70, 22]).

Regarding RL2, BPMN was the only language used to represent the structured parts of business processes, while several other declarative languages were used to represent the unstructured parts (i.e., [68], [78], [73], [79], [65], [59], [60], [46]). The proposed HBPRs in this case were represented using the combination of two static artifacts (i.e., a process model and a textual description).

In addition, several approaches in both RL1 and RL2 abstracted from particular language specifications and rather proposed generic approaches that can be adapted to a wide range of declarative and imperative languages. In RL1, some articles (i.e., [37, 25, 58, 76, 77]) propose hybrid approaches combining generic imperative constructs with declarative constraints. The proposed HBPRs were all represented as static artifacts i.e., process models. Alternatively, in RL2, some articles (i.e., [71, 66, 72]) restricted the modeling of imperative specifications to BPMN , while they still abstracted from choosing a particular language to model declarative specifications. Nevertheless, all the resulting HBPRs assume the combination of two static artifacts i.e., a process model to represent imperative specifications and textual descriptions to represent declarative specifications. Therefore, the use of these approaches is limited to only declarative languages which are conventionally represented in a textual format.

By looking at the different combinations of artifacts shown in Table 3, one can notice that none of the proposed HBPRs comprises a dynamic artifact. On that matter, one can argue that none of the covered approaches has deployed input from event logs to support HBPRs. This limitation is further discussed in Section 6.

5.5. Taxonomy

This section presents a descriptive taxonomy (answering RQ5) based on the outcome of the combinations of the artifacts and the languages presented in Section 5.4. Process artifacts can be categorized according to their inherent and visual features (cf. Section 2.3). The inherent features can be used to discern the formality and the language paradigm characteristics of the languages combined in a HBPR. In terms of formality, some HBPRs in the literature *(1)* combine two languages having a formal syntax and formal semantics, whereas other HBPRs *(2)* combine a language having a formal syntax and semantics with a language having a semi-formal syntax or semantics. Another set of HBPRs *(3)* uses a language having a formal syntax and semantics together with a language having an informal syntax and semantics. Table 4, categorizes the literature articles according to their language formality[6]. In terms of the language paradigm, HBPRs can be grouped as previously shown in Table 3.

The visual feature can be used to discern the types of the artifacts combined in the proposed HBPRs. As mentioned in Section 2, HBPRs can be divided into *(a)* hybrid languages and *(b)* hybrid process artifacts. In the literature, hybrid languages are composed in a single static artifact, whereas hybrid process artifacts are composed using multiple static and interactive artifacts. By looking closely at the articles describing hybrid languages, two different types of structures emerge: *hierarchical structures* and *mixed structures*. With hierarchical structures, process models are fragmented into sub-processes or so so called “Pockets of Flexibility” [37] where each subprocess or pocket can be modeled using a declarative or an imperative language. Such a decomposition reduces the complexity of the hybrid representation and allows for s better reusability of the existing model fragments [24]. Alternatively, *mixed structures* allow combining declarative and imperative

---

[6] Approaches abstracting from combining particular language specifications are not covered by the formality grouping.

Motivations	Articles in RL1	Articles in RL2
(a) Process flexibility	[37], [25], [76], [77],[58], [50], [75], [51], [24], [52],	[71], [66], [74], [65], [59], [72]
(b) Separating and integrating business rules		[71], [66], [74], [65], [59], [72], [60]
(c) Conciseness and preciseness of hybrid languages	[50], [51], [24], [52]	
(d) Understandability	[52], [69], [16], [67], [45], [70], [22]	[68], [78], [73], [79], [74], [65], [59], [46], [21], [60]
(e) Maintainability	[69], [16], [37], [25], [70], [22]	[66], [59], [72], [21], [60]
(f) Improving the PPM	[69], [16], [67], [70], [22], [21]	

Table 2: Summary of the motivations behind the HBPRs proposed in the literature.

		Languages														Artifacts		
		Imperative						Declarative								Static		Interactive
	Work	BPMN	YAWL	Petri net	C.Petri net	I-NL	Lang. Indep/Unspecified	Declare	DCR	CMMN	C-NL	R2ML	SBVR	XTT2	Lang. Indep./Unspecified	Text.Des.	P.Model	Guided Sim.
RL1	[75]		x					x									x	
	[8]		x					x									x	
	[50]				x			x	x								x	
	[24]			x				x									x	
	[52]			x				x									x	
	[51]	x						x									x	
	[67]					x		x								x	x	
	[45]					x		x								x	x	
	[69]						x	x									x	x
	[16]						x	x									x	x
	[70]						x		x								x	x
	[22]						x		x								x	x
	[21]					x			x		x					x	x	
	[37]						x								x		x	
	[25]						x								x		x	
	[58]						x								x		x	
	[76]						x								x		x	
	[77]						x								x		x	
RL2	[68]	x											x			x	x	
	[78]	x											x			x	x	
	[73]	x												x		x	x	
	[79]	x												x		x	x	
	[74]	x										x				x	x	
	[65]	x										x				x	x	
	[59]	x										x				x	x	
	[60]	x								x							xx	
	[46]	x									x					x	x	
	[71]	x													x	x	x	
	[66]	x													x	x	x	
	[72]	x													x	x	x	

Table 3: List of the languages and artifacts combined in the HBPRs proposed within RL1 and RL2.

		Formal and Formal		Formal and Semi-formal	Formal and Informal		Generic Language	
		RL1	RL2	RL2	RL1	RL2	RL1	RL2
Hybrid Language	Hierarchical Structure	[8], [75], [24]					[37], [76], [77], [25], [58]	
	Mixed Structure	[50], [51], [52], [60]						
Hybrid Artifacts	P. Model and T. Descriptions		[73], [79]	[68], [78],[74], [65], [59]	[67], [45], [21]	[46]		[71], [66], [72]
	P. Model and Guided Sim.						[69], [16], [70], [22]	

Table 4: Categorization of the literature articles based on their language formality and visual features. *"Formal and Formal"* refers to the combination of two languages having a formal syntax and formal semantics. *"Formal and Semi-formal"* refers to the combination of a language having a formal syntax and semantics with a language having a semi-formal syntax or semantics. *"Formal and Informal"* refers to the combination of a language having a formal syntax and semantics with a language having an informal syntax and semantics.

languages within the same process or sub-process. This way, languages from different ends of the imperative–declarative paradigm spectrum can be fully mixed to represent both imperative and declarative specifications in a compact manner. Although this approach is not very common in the literature some articles (i.e., [50, 51, 52, 60]) have used mixed structures in the design of their hybrid languages. The structures of hybrid languages identified in the literature complement the characteristics of hybrid languages introduced in Section 3.1.

Hybrid process artifacts have been represented differently in the literature. Some articles combine two static process artifacts, which are represented as a combination of a process model with textual descriptions. Other approaches combine a static artifact with an interactive one, namely, a process model with a guide simulation. Table 4 categorizes the literature articles based on their visual features, and Figure 15 summarizes the proposed taxonomy graphically.

5.6. Maturity

This section evaluates the maturity of the proposed approaches (answering RQ6). To this end, a set of maturity criteria has been defined according to the recommendations of modeling experts. First, the degree of formalization evaluates the extent to which the proposed approaches can be used to design a HBPR consistently. Secondly, the availability of implementation provides indications about how swiftly they can be applied in realistic settings. Finally, empirical evaluations provide an assessments of the proposed approaches under different conditions. Section 5.6.1 compares the articles based on their degree of formalization. Section 5.6.2 provides details about the implementations proposed in the literature and compares them. Section 5.6.3 describes the results of the empirical evaluations conducted in the literature and provides a comparison for them.

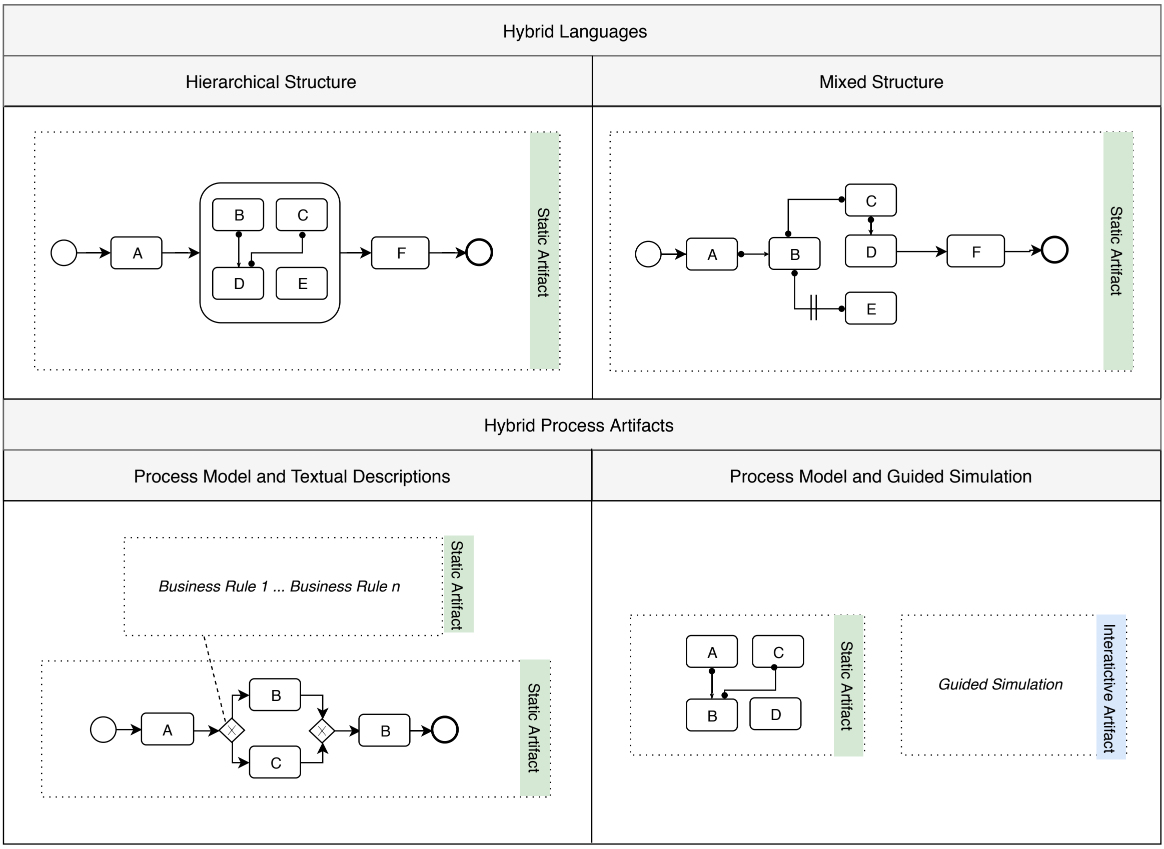


Figure 15: Summary of the descriptive taxonomy. Black edges with arrow head (adapted from BPMN) are used to illustrate the control flow expressed by imperative languages, whereas black edges with bullet head (adapted from Declare) are used to illustrate declarative constraints. Process Models and guided simulations have a formal syntax and formal semantics, whereas business rules can have a formal, semi-formal or informal syntax or semantics.

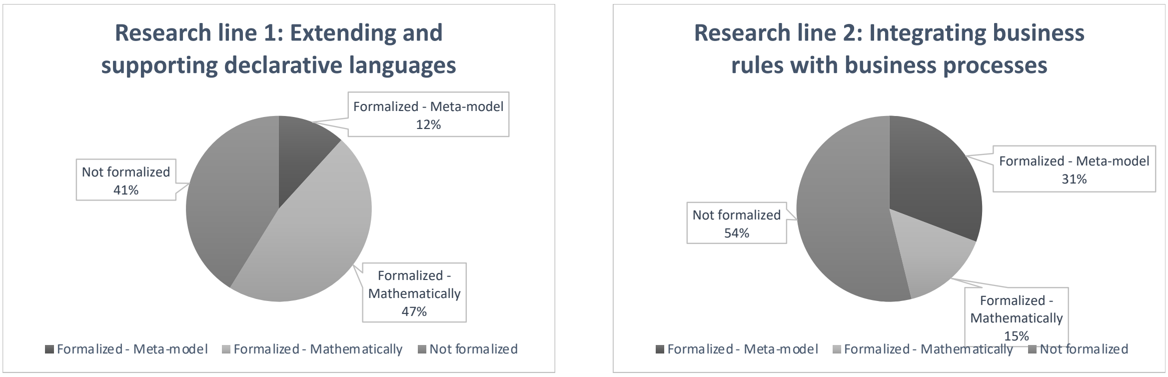
Deg. of Formalization	Publications Refs.
Meta-model	[74], [65], [59], [69], [16], [60]
Mathematical	[77], [75], [51], [24], [52], [71], [67], [45], [25], [58]
No Formalization	[37], [76], [8], [50], [68], [78], [73], [79], [66], [72], [46], [22], [70], [21]

Table 5: Degrees of formalization of each of the HBPRs proposed in the literature.

5.6.1. *Formalization*

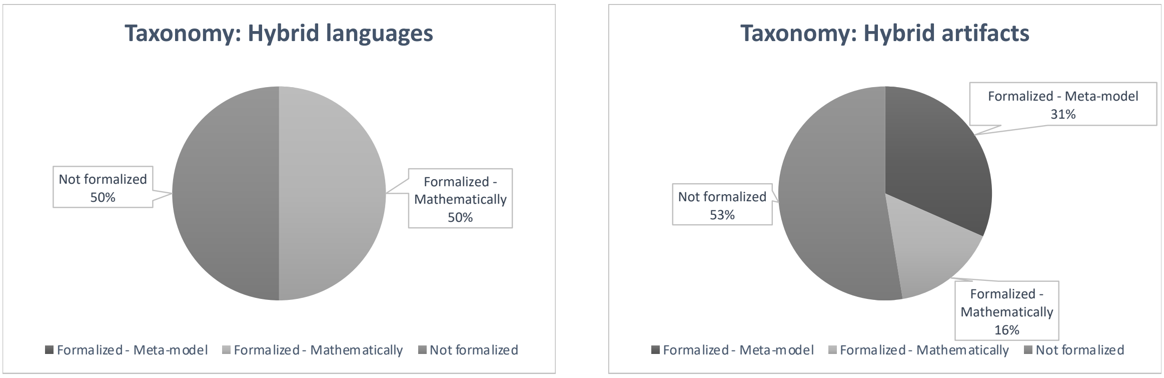
The approaches introduced in the literature are described with varying levels of formalization. Indeed, part of the proposed approaches use meta-models to describe the components of their HBPRs. Although meta-models might not be enough to formally describe the proposed approaches, their abstraction allows conveying the overall idea and provides a moderate understanding to the reader. However, as a mathematical formalization is missing for these approaches, side issues and misinterpretations might be encountered during their deployment. Besides, several authors in the literature have provided a mathematical formalization to describe their approaches. With this regards, the proposed formalizations provide an overarching understanding of the approaches and mark their readiness for deployment in realistic settings. Another portion of the literature did not provide any formalization to their approaches. By inspecting the articles with none formalized approaches, one can notice that most of them were either part of new initiatives, intermediate work, evaluations or approaches proposed in ad-hoc contexts (i.e., for specific case studies). Table 5 summarizes the degrees of formalization of each of the approaches proposed in the literature.

Figure 16 compares the formalization of the approaches based on the research lines categorization and the taxonomy introduced in Sections 5.2 and 5.5 respectively. By comparing the two research lines, one can notice that nearly half of the ap-



(a) Distribution of articles of RL1 (cf. Section 5.2.2) according to their formalization.

(b) Distribution of articles of RL2 (cf. Section 5.2.3) according to their formalization.



(c) Distribution of articles proposing a hybrid language (cf. Section 5.5) according to their formalization.

(d) Distribution of articles proposing a hybrid process artifact (cf. Section 5.5) according to their formalization.

Figure 16: Distribution of the articles according to their formalization based on different categorizations.

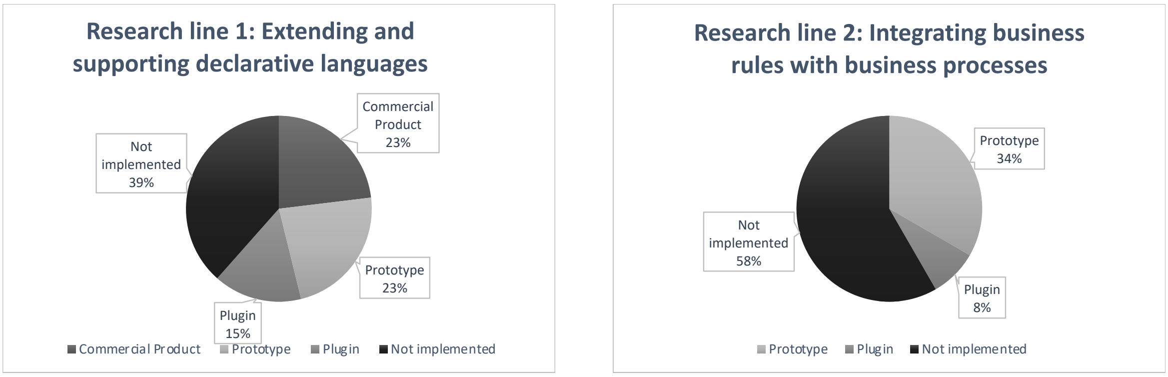
proaches proposed within RL1 and RL2 are formalized. Looking at the different types of formalization, one can see a tendency to formalize approaches in RL1 mathematically, whereas more approaches in RL2 are formalized using a meta-model. By comparing the approaches based on the proposed taxonomy (i.e., hybrid languages, hybrid artifacts), one can notice a balanced distribution between none-formalized and formalized approaches in both research lines. However, none of the approaches in RL1 has been formalized using a meta-model, whereas, nearly two-thirds of approaches in RL2 have been formalized using a meta-model. These insights suggest that the approaches in RL1 and the approaches proposing a hybrid language are the candidates to represent HBPRs consistently, which in turn raise the need to provide mathematical formalizations for the approaches in RL2 and for the approaches proposing a hybrid artifact.

5.6.2. *Availability of Implementation*

The availability of implementation is another important aspect allowing to assess the maturity of the proposed HBPRs. Information about the implemented approaches including the tool name, type, parent framework and a reference to the tool are shown in Table 6. This information provides clear insight into the maturity of the proposed approaches in terms of their implementation characteristics. For instance, the implementation type allows discerning whether the tool is a prototype, a plugin or a commercial product. Furthermore, the implementation type can also provide indications about the tool integration in industry, as commercial products are more likely to be used in industry compared to prototypes or plugins.

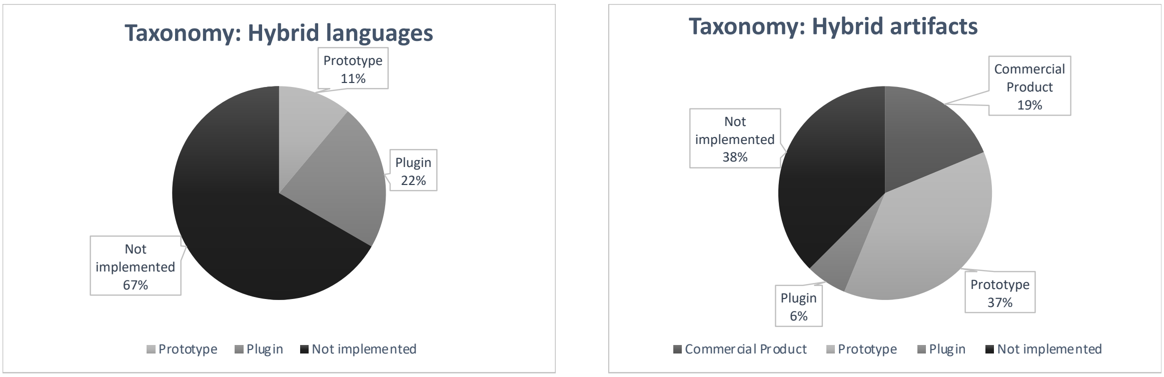
Among the 30 articles found in the literature, 18 comprise an implementation. As shown in Table 6, most of the implemented approaches were either prototypes or plugins, whereas only 3 approaches are available as part of a commercial product. Furthermore, as prototypes were mostly used as a proof of concept, some articles have not shared their implementation source code in their publications while other implementations have been discontinued.

By comparing the approaches proposed within the two research lines identified in Section 5.2 (cf. Figures 17a and 17b), it is visible that more approaches have been implemented in RL1 compared to RL2. In addition, by considering the devel-



(a) Distribution of articles of RL1 (cf. Section 5.2.2) according to their implementation type.

(b) Distribution of articles of RL2 (cf. Section 5.2.3) according to their implementation type.



(c) Distribution of articles proposing a hybrid language (cf. Section 5.5) according to their implementation type.

(d) Distribution of articles proposing a hybrid process artifact (cf. Section 5.5) according to their implementation type.

Figure 17: Distribution of the articles according to their implementation type based on different categorizations.

Publications	Tools Name	Implementation Type	Parent Frameworks	References
[22]	DCR swimlanes	Commercial Product	DCR Solutions	`http://dcrgraphs.net`
[67, 45]	Declare Execution Environment	Prototype		`http://processmining.be/declareexecutionenvironment`
[59]	vBPMN	Prototype	jBoss Drools 5.1	N/A
[66]	Unamed	Prototype	Aqualogic BPM Studio and ILOG business rules engines	N/A
[71]	Unamed	Prototype	IBM WebSphere Integration Developer/ Process Server	N/A
[73, 79]	Oryx-HQEd	Plugin	Oryx Editor	`https://ai.ia.agh.edu.pl/wiki/hekate:start`
[70]	DCR Simulation Tool	Commercial Product	DCR Solutions	`http://dcrgraphs.net`
[21]	Process Highlighter	Commercial Product	DCR Solutions	`http://dcrgraphs.net`
[60]	The knowledge work designer	Prototype	ADOxx	`http://adoxx.org`
[37, 25]	Chameleon	Prototype	FlowMake	Discontinued
[75, 8]	Declare service	Plugin	YAWL environment	cf. [81]
[50]	Unamed	Plugin	CPN Tools	`http://cpntools.org` cf. [82]
[69, 16]	TDMS	Prototype	Cheetah, Declare Framework	`http://www.zugal.info/tdms`

Table 6: Information about the implementations proposed in the literature.

opment of commercial products, one can claim that RL1 comprises mature implementations which can be adopted in industrial settings. When comparing the proposed approaches based on the taxonomy in Section 5.5 (cf. Figures 17c and 17d), one can notice that the largest portion of hybrid languages has not yet been implemented and the rest were implemented either as prototypes or plugins. Regarding hybrid artifacts, a large portion has already been implemented as commercial products, prototypes and plugins. This insight denotes the maturity of approaches implementing hybrid artifacts and raises the need for providing a tool-support for the existing hybrid languages since some of them have been already formalized (cf.Section 5.6.1).

5.6.3. *Empirical Evaluation*

The availability of empirical evaluations is considered as an important factor to evaluate the maturity of the approaches presented in the literature. Among the 30 articles in the literature, only 5 HBPRs developed in 8 articles i.e., [46, 69, 16, 67, 45, 70, 22, 21] were evaluated in the literature. As shown in Table 4, these articles correspond to approaches proposing a hybrid artifact, in particular, approaches combining a process model with textual descriptions and approaches combining a process model with a guided simulation. Table 7 provides an overview about the approaches evaluated in the literature.

The TDM approach [69, 16] was evaluated in [23] and [83], the evaluation has covered two empirical studies. The first study covered 8 participants and investigated the extent to which the proposed HBPR helps to foster the communication with domain experts and IT specialists during the PPM. In this experiment, the communication between the process stakeholders during the PPM was recorded. Then following, the CoPrA approach [84], the verbal data was transcribed and coded according to a specific coding scheme. Additionally, video recordings of the modeling sessions were also collected to perceive the context of the verbal data. A qualitative analysis of the coded data shows that test cases were accepted by the participants as communication channel and the HBPR contributes to a better understandability of declarative process models. The second study has covered 12 participants and investigated the maintainability of the proposed representation. In this study, the TDM framework and the Cheetah experimental platform [85] were used to track the PPM and to assess the quality of the obtained process models. In addition, questionnaires were deployed to measure the cognitive load and the quality of the process models as perceived by the participants by the end of the modeling session. The results of a quantitative analysis demonstrate that the proposed representation improves the maintainability of declarative process models, lowers the cognitive load, and increases the perceived model quality.

The approach extending a declarative process model with textual descriptions of the hidden dependencies between the Declare relations [67, 45] was evaluated in two studies. A first study was reported in [67] with 95 participants then extended in a second study to cover 146 participants in [45]. In this work, the authors investigate the impact of the proposed HBPR on model understandability by scrutinizing the effect of adding an extra layer of textual descriptions to declarative process models. To this end, the authors used Declare Execution Environment (cf. Table 6) to record important interactions (i.e., opening a dependency graph visualization) and response time. Furthermore, the authors use questionnaires to evaluate the participants comprehension and self-assessment of cognitive load. The results of quantitative and qualitative analyses demonstrate that the use of textual descriptions contributes to an enhanced understandability, reduced mental effort and response time when dealing with a HBPR compared to a declarative process model representation.

The understandability of HBPRs where linked rules are combined with imperative process models was evaluated in [46]. This representation illustrates the common representation of HBPRs proposed in RL2 (cf. Section 5.2.3). The study has covered 58 participants. In order to investigate the impact of integrating business rules in process models, an Eclipse RCP application[7] illustrating a HBPR was developed. Furthermore an eye tracking device was deployed to record the participants gaze data during the experiment, which in turn, were used to derive the total fixation duration (i.e., sum of the duration of all fixations on a specific area of the stimulus [86]). As a performance measure, the response times of participants were used. In addition, cognitive load was measured objectively using eye fixation data [87] and subjectively using the participants' self-assessments of the perceived mental effort. The results of a quantitative analysis show that participants had a higher comprehension accuracy and lower mental effort dealing with a HBPR compared to an ordinary representation where rules are separated from the process model. These insights demonstrate that the combination of an imperative process model with linked rules is associated with an enhanced understandability.

The combination of DCR graphs, a flow-based representation of the control-flow and a simulation tool [22, 70] was evaluated by A. Andaloussi et al. [44]. The authors deploy a HBPR combining a DCR process model with a guided simulation and a swimlane illustrating the flow-based representation. The initial study has covered 10 participants (university students and municipal employees). In order to investigate the understandability of the proposed HBPR, the authors examined the distribution of attention on the different artifacts using a set of eye tracking fixation-based measures [88] including fixation count (i.e., number of fixations on specific area of the stimulus [86]) and total fixation duration. Furthermore, the authors identified the common reading patterns of participants following a process mining based approach proposed in [89]. With this regard, fixation data were converted to event logs, then a process discovery technique [49] was used to infer the implied attention maps. The results of a qualitative analysis demonstrate an unbalanced distribution of attention over the different artifacts and denote the presence of different user profiles exhibiting different reading patterns of the proposed representation. A follow up study was conducted in [43]. In this work, the authors triangulated the subjective insights obtained from the retrospective think-aloud sessions with the objective data recorded by the eye tracking device. The follow up study, which has covered 15 participants with different backgrounds, highlighted the benefits and the challenges associated with using each of the HBPR artifacts individually, and investigated the way users with different backgrounds engage with each of the deployed artifacts. In addition, the study explored the different reading patterns associated with different types of tasks. The results show that the deployment of a single artifact is not enough to provide an overarching understanding for domain experts and IT specialists when dealing with different tasks, which in turn motivate the use of HBPRs.

The process highlighter proposed by Lopez et al. [21] was evaluated by A. Andaloussi et al. [48]. The study has covered 17 participants including employees at a Danish municipality and university students. In this work, the authors investigated the potential support offered by a HBPR during the PPM. The

---

[7] See `https://wiki.eclipse.org/Rich_Client_Platform`

Articles	Ev. Ref.	# Par.	Ev. type	Instruments	Measurements	Research Aspects
[46]	[46]	58	Quan.	RCP app, eye tracking, questionnaires	Comprehension accuray, response time, total fixation duration	Effect of integrating BR with BP
[69, 16]	[83]	8	Qual.	Audio and video recordings, questionnaires	Coding of transcribed verbal data	Communication during the PPM
		12	Quan.	TDM framework, Cheethah experimental platform	Cognitive load, perceived quality, quality of model	Maintainability and model quality
[67, 45]	[45]	146	Both	Declare Execution Environment, questionnaires	User interactions, response time, self-rating of cognitive load	Impact on comprehension difficulty
[70, 22]	[44, 43]	15	Quan.	Eye tracking, user Interactions	Fixation-based measures, coding of transcribed verbal data	Attention, reading patterns, use of a HBPR artifacts
[21]	[48]	17	Quan.	User Interactions	User Interactions, coding of transcribed verbal data	Benefits of using a HBPR during the PPM

Table 7: Summary of the evaluation findings. The outcomes of the studies are described in Section 5.6.3. New acronyms: Par. (Participants), Ev. (Evaluation), Quan. (Quantitative), Qual. (Qualitative), BR (Business Rules) and BP (Business Processes).

triangulation of the subjective insights obtained from the think-aloud data and the objective insights obtained from the user interactions highlights the support provided by the proposed HBPR during the PPM and hints toward an enhanced quality of process models w.r.t to alignment, traceability and documentation of the process specifications.

5.7. Application Domains

This section reports the different application domains of HBPRs referred in the literature (answering RQ7). The approaches proposed in the literature target dynamic business environments characterized by continuous changes in customer's attitudes and regulations. The authors in the literature illustrate the applicability of their approaches in several domains either by investigating specific case studies or by providing realistic examples illustrating the challenges faced in industrial settings.

The literature shares the same requirements for flexible, understandable and maintainable systems able to adjust for changing customers needs in different domains. As shown in Table 8, the literature covers a variety of application domains such as Health Care, Education, Customer Relation Management (CRM), Web Content Management (WCM), Human Resources Management (HRM), Consultancy, Logistics, Insurance, Banking, and Citizens Services. Moreover, the application of some approaches in the literature is not only limited to specific domains but could also be adapted in designing any HBPR sharing similar needs and motivations as the ones discussed in the literature.

Application Domains	References
Auctioning Service	[78]
Book Store Management	[65]
Consultancy Service	[54]
Funding Applications	[24]
Governement Citizens Service	[66], [22], [21]
Health Care	[50], [74], [72] [51], [37], [25], [58], [76], [77]
Education	[37], [25], [58], [76], [77], [60]
CRM	[51], [37], [25], [58], [76], [77]
WCM	[76], [77]
Human Ressource Mangement	[75]
Liability Insurrance Processes	[73], [79]
Loan Application	[2]
Order Delivery/Cash Processes	[52], [71]
Ship Engine Maintainance	[59]
Generic HBPR	[8], [68], [46], [69], [16], [67], [45], [70]

Table 8: Application Domains of the approaches proposed in the literature.

6. Discussion

The analysis presented in Section 5 enabled answering the different research questions of this work. This section discusses the results of the analysis with a twofold purpose. On the one hand, it highlights the important findings in the literature (cf. Section 6.1). On the other hand, it provides a research agenda based on the key findings to guide the emergence of HBPRs (cf. Section 6.2).

6.1. Key Findings

The research lines identified in Section 5.2 discern the contexts where hybrid representations have been proposed. Although both research lines combine languages and artifacts and share similar motivations (e.g., flexibility, understandability and maintainability) the underlying approaches have evolved within their own cluster. In the context of the HBPRs covered in this study, the work of Pesic et al. [8] seems to have the most widespread traction. However this traction is mostly due to the specification of the Declare language, which was cited by all the approaches proposing a hybrid representation including Declare. Besides that, the approaches proposed by Sadiq et al. [37] and Westergaard and Slaats [50] are the ones with the most widespread traction within RL1. Indeed, the two structures characterizing hybrid languages (cf. Section 5.5) have been initiated in these two publications. Namely, Sadiq et al. [37] proposed the hierarchical structure to combine hybrid languages, whereas Westergaard and Slaats [50] proposed the mixed structure. Most of the upcoming publications about hybrid languages have taken inspiration from either of the two approaches. In RL2, the cross-referencing between the different articles is lower compared to RL1. Nevertheless, the modeling patterns used to extract business rules from business processes proposed by Graml et al. [71] were the primary source of inspiration for other similar approaches.

Looking at the combined languages and artifacts in Section 5.4, it is clear that most of the existing hybrid languages combine Declare language with imperative languages, whereas, most of the hybrid process artifacts combine declarative artifacts with process models in BPMN. The descriptive taxonomy introduced in Section 5.5 extends the conceptual framework instantiated in Section 3 by discerning the characteristics of the existing HBPRs. The taxonomy shows that none of the proposed HBPRs comprises a dynamic artifact. Indeed none of the covered approaches has deployed traces from event logs to provide a dynamic visualization as part of a HBPR. This is due to the descriptive nature of the proposed taxonomy which emphasizes only the characteristics of the HBPRs covered by the

SLR search. In addition, as mentioned in Section 4.5, this study emphasizes only the modeling of HBPRs, thus the approaches mining HBPRs were excluded, which might explain the lack of approaches incorporating a dynamic artifact. Nevertheless, as explained in Section 2.3, the distinction between dynamic and interactive artifacts is rather clear, which, in turn, motivates its placement into the proposed conceptual framework.

The maturity aspects scrutinized in Section 5.6 provide indications about the ability of the proposed HBPRs to be integrated in industrial settings. In terms of formalization, more approaches proposing hybrid languages are formalized compared to those proposing hybrid artifacts. Regarding the availability of implementation, an inverse pattern can be observed: implementations are more common for hybrid artifacts than hybrid languages. We conjecture that this may be the case because research on hybrid languages tends to be of a more theoretical nature and therefore a formal treatment of the language is expected, whereas research on hybrid artifacts tends to focus more on questions of understandability, which are best demonstrated empirically through the implementation of tools. In terms of the maturity of implementation, except for a few commercial tools, most of the proposed approaches within both research lines present prototypes or plugins mainly as a proof of concept. However, in order to reach the industrial market and to ensure a positive impact on the development of HBPRs, more robust implementations are required. In terms of empirical evaluation, few approaches have been evaluated so far and these cover only hybrid process artifacts.

6.2. Research Agenda

This section discusses a research agenda to delineate the directions for the up-coming research. In order to promote the use of HBPRs, it is necessary to consider the entire HBPR development-cycle, which includes the following phases: *(a)* design, *(b)* modeling and *(c)* evaluation.

Design: In the past two decades, a set of approaches and methodologies has been proposed to design HBPRs. Still, little is known about the synergies and overlaps between the languages composing these hybrid representations. Besides a handful of articles looking into the representation capabilities of hybrid process artifacts integrating business rules with business processes (e.g., [90]), most of the other combinations remain unexplored. On that matter, it is necessary to conduct more ontological analyses questioning the overlap between the existing modeling languages and investigating how different languages can semantically complement each other to derive concise representations of business processes.

It is also crucial to consider the human factor during the design of HBPRs as Lindland et al. [91] said “not even the most brilliant solution to a problem would be of any use if no one could understand it”. In the field of process modeling, cognitive psychology has been deployed to compare the visual support offered by different process modeling languages [92]. For instance, Figl et al. [92] compared the control-flow constructs of several languages (e.g., YAWL, BPMN, EPC) according to a subset of the visual design principles introduced by Moody [93] as part of the physics of notations framework. Namely, the covered languages have been compared based on their representational clarity (i.e., the fit between the graphical symbol representing a construct and the semantic concepts referring to it), perceptual discriminability (i.e., the ease to distinguish between the graphical symbols of a language), perceptual immediacy (i.e., the extent to which a graphical symbol can provide a cue to its meaning), visual expressiveness (i.e., the use of visual variables such as shape, size and color in a language) and graphic parsimony (i.e., the graphical complexity of a language). The physics of notations framework provides a comprehensive set of visual design principles allowing to investigate the cognitive effectiveness of single languages – but also hybrid languages and hybrid process artifacts. This in turn could be used to derive hybrid representations with an increased cognitive support.

Modeling: The quality of process models has been extensively investigated in the field of process modeling. Accordingly, several guidelines aiming at enhancing the quality of process models have emerged. The Seven Process Modeling Guidelines (7PMG) [94] and the SEQUAL framework [95] comprise a set of quality aspects defining the criteria for understandable process models. These criteria have emerged as a result of several empirical studies investigating the reading and the modeling of imperative process models, thus, their applicability remain questionable for declarative process models. Since hybrid representations combine languages from both paradigms, more research is required to *(1)* differentiate the guidelines which can be applicable to languages from both paradigms (e.g., verb-object naming of activities, number of elements in the model) and *(2)* derive new guidelines covering the aspects specific to declarative languages (e.g., the placement of entry-point and exit-point activities in a declarative process model). Afterwards, it is necessary to evaluate the applicability of these guidelines on hybrid representations and refine them accordingly to fit the intended purpose. Considering the variety of combinations of languages and artifacts proposed in the literature, the new guidelines should cover the different classes of HBPRs presented in the descriptive taxonomy (cf. Section 5.5).

The modeling of HBPRs is also constrained by the quality of the tools supporting process modeling. In this context, it is important to guide users toward using what is best in a context specific manner. By learning from users’ behavior and the contextual information available at run-time, an adaptive system can be developed to provide a set of recommendations allowing to enhance the interactions with the HBPR. In this direction, initiatives have been made in the field of process modeling to discern the different phases associated with the modeling of BPMN process models (i.e., problem understanding, method finding, modeling and reconciliation) based on eye tracking and user interaction data [96]. Hence, similar approaches could be developed to identify the features defining the different modeling phases of a hybrid representation, which in turn could be used to provide a phase-specific modeling support at run-time.

Evaluation: Process modeling languages have been widely evaluated with regards to their understandability (e.g., [97, 98, 99]), maintainability (e.g., [100]) and modeling (e.g., [101]). However, when it comes to hybrid approaches, there is not much empirical work yet. Indeed most of the existing empirical studies are limited to the understandability of hybrid process artifacts, mainly those combining a process model with a textual annotation or a guided simulation, therefore, it is necessary to extend the evaluations to cover all the other classes of hybrid representations. Empirical evaluations should not be limited to the understandability of hybrid representations but should also cover other perspectives such as the modeling, the maintainability and the communication support offered by hybrid representations. The evaluation of the process highlighter presented in [48] is one among the few studies investigating the modeling using a HBPR and reporting some modeling patterns, even though, the provided insights remain of exploratory nature. To understand the way users engage with modeling tasks using HBPRs, more confirmatory studies are required. In this vein, HBPRs could be compared in terms of the perceived quality of process models and their alignment with the process specifications. In addition, physio-psychological measures can be used to estimate the cognitive load associated with the use of the different representations.

The support for better maintainability is another aspect to be investigated. The empirical evaluation of the maintainability of the TDM [83] is a starting point in this direction. However, due to the limited number of participants and the subjectivity of the used measures, the results cannot be generalized. Besides that, the maintainability of other classes of hybrid representations has to be evaluated as well. Part of the literature (cf. Section 5.3) claims the maintainability support offered by hybrid representations, however, little is known about the proper approach to maintain overlapping artifacts where certain information might be redundant.

The communication support offered by hybrid representations is another aspect to be investigated in the literature especially when dealing with hybrid process artifacts. In that respect, it is important to investigate the ability of hybrid process artifacts to bridge the communication gap between domain experts and IT specialists. An existing study hints towards this kind of support [23]. However, due to the lack of participants, no strong inferential statistics could be made. An in-depth understanding of this aspect would have a strong impact on the development of PAIS systems.

A large portion of existing empirical evaluations use students as subjects to evaluate their approaches, which, in turn, limits the validity of the results to academia. However, in order to be able to generalize the obtained results, it is crucial to go beyond academia and explore the use of HBPRs in industrial and administrative settings. Among the empirical studies to be cited in this direction is the study by [43], where the authors explored the understandability of HBPRs in both academic and administrative settings. In this work, the authors were able to spot considerable differences w.r.t the interactions of students and municipal employees with HBPRs. The results provide preliminary insights which can serve as a basis for future research in this direction.

A great variety of measurements are deployed to assess the interaction of humans with software artifacts. From basic questionnaires to advanced technologies (e.g., eye tracking, electroencephalography) these techniques have proven their efficacy in several user experience studies (e,g., [102]. When considering HBPRs, the deployment of such measurements remains limited, except for a few studies which report eye-tracking related insights (e.g., [46, 44]), the use of physio-psychological measurements to evaluate HBPRs remains very limited. Understanding the human cognitive processes and discerning the different strategies and patterns when reading, modeling and maintaining HBPRs is therefore vital to develop robust hybrid representations.

7. Threats to Validity

The validity of this SLR is subject to some threats particularly related to completeness, selection bias and the reliability of the automated tools used through the data collection and analysis. To ensure the search completeness, generic keywords were appended to the search string after a series of string refinements and search iterations. Moreover, the results of the backward search and forward search allowed identifying the extra articles that were not identified during the main literature search. To avoid any selection bias the articles were selected systematically according to a set of clear inclusion and exclusion criteria. Nevertheless, slight selection bias might be subjectivity introduced during the initial selection process as some articles did not clearly describe and motivate the aim of their approaches. Furthermore, articles published after September, 2018 were not covered by the search as the literature search was completed by that time. The search engines deployed to retrieve relevant literature constitute another potential threat to validity. With this regard, a simple and comprehensive search string was formulated using a single logical operator as some search engines do not support complex queries. In addition, the search process was fragmented to cover each publication venue individually in order to ensure a consistent search through all the relevant venues. Finally, the backward search was performed automatically by extracting and parsing the references cited in each of the selected articles, which in turn, were used to generate the graph visualization depicting the different research lines. A potential risk of this approach is associated with the inability to parse some references which can impact of the validity of the presented visualization.

8. Conclusion

This work proposes a conceptual framework and summarizes the outcome of a SLR about HBPRs. At a first stage a unified terminology is defined and the characteristics discerning hybrid languages and hybrid process artifacts are presented. Afterward a SLR is conducted to explore the existing literature. The analysis of the SLR findings allowed identifying the characteristics of existing HBPRs and the motivations driving their emergence. The data extracted from the literature allowed the identification of two research lines i.e., one aiming at extending and supporting declarative languages and another aiming at integrating business rules in business processes.

For both research lines, the underlying publications were scrutinized closely. To this end, the combined representations were analyzed and grouped to derive a descriptive taxonomy. In addition, the maturity of the proposed approaches was profoundly examined. In addition, the common application domains where the use of these approaches is beneficial were identified and presented at the analysis.

The discussion of the findings reveled important insights about the results of the literature research, and provided a comprehensive research agenda tracing out the directions for future work while considering each phase of the HBPR development-cycle.

Finally, The overall contributions of this study allowed developing a deepened understanding of HBPRs. The outcome of this study will contribute to the development of new HBPRs and will serve as a basis to be systematically updated with upcoming research.

References

[1] J. Becker, M. Rosemann, C. von Uthmann, Guidelines of Business Process Modeling (2000) 30–49`arXiv:9780201398298`, `doi:10.1007/3-540-45594-9_3`.
URL `http://link.springer.com/10.1007/3-540-45594-9_3`

[2] J. Pinggera, T. Porcham, S. Zugal, B. Weber, LiProMo-Literate process modeling, CEUR Workshop Proceedings 855 (2012) 163–170.

[3] B. Axenath, E. Kindler, V. Rubin, AMFIBIA: A meta-model for the integration of business process modelling aspects, International Journal on Business Process Integration and Management 2 (2) (2007) 120–131.

[4] M. Dumas, W. M. P. van der Aalst, A. H. M. ter Hofstede (Eds.), Process-Aware Information Systems, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2005. `doi:10.1002/0471741442`.
URL `http://doi.wiley.com/10.1002/0471741442`

[5] W. M. P. van der Aalst, Process-Aware Information Systems: Lessons to Be Learned from Process Mining, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 1–26. `doi:10.1007/978-3-642-00899-3_1`.
URL `https://doi.org/10.1007/978-3-642-00899-3_1`

[6] O. M. G. OMG, Business process modeling notation v 2.0.
URL `https://www.omg.org/spec/BPMN/2.0/`

[7] T. Murata, Petri nets: Properties, analysis and applications, Proceedings of the IEEE 77 (4) (1989) 541–580. `doi:10.1109/5.24143`.

[8] M. Pesic, H. Schonenberg, W. M. van der Aalst, DECLARE: Full Support for Loosely-Structured Processes, in: 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007), IEEE, 2007, pp. 287–287. `doi:10.1109/EDOC.2007.14`.
URL `http://ieeexplore.ieee.org/document/4384001/`

[9] T. T. Hildebrandt, R. R. Mukkamala, Declarative Event-Based Workflow as Distributed Dynamic Condition Response Graphs, Electronic Proceedings in Theoretical Computer Science 69 (2011) 59–73. `arXiv:1110.4161`, `doi:10.4204/EPTCS.69.5`.
URL `http://arxiv.org/abs/1110.4161v1`

[10] D. Fahland, D. Lübke, J. Mendling, H. Reijers, B. Weber, M. Weidlich, S. Zugal, Declarative versus Imperative Process Modeling Languages: The Issue of Understandability, in: Proceedings of EMMSAD, Springer, 2009, pp. 353–366.

[11] S. Goedertier, J. Vanthienen, F. Caron, Declarative business process modelling: Principles and modelling languages, Enterp. Inf. Syst. 9 (2) (2015) 161–185. `doi:10.1080/17517575.2013.830340`.
URL `http://dx.doi.org/10.1080/17517575.2013.830340`

[12] M. Reichert, B. Weber, Enabling Flexibility in Process-Aware Information Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. `doi:10.1007/978-3-642-30409-5`.
URL `http://link.springer.com/10.1007/978-3-642-30409-5`

[13] H. A. Reijers, T. Slaats, C. Stahl, Declarative modeling - An academic dream or the future for BPM?, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 8094 LNCS (2013) 307–322. `doi:10.1007/978-3-642-40176-3_26`.

[14] M. Pesic, Constraint-Based Workflow Management Systems : Shifting Control to Users, phd thesis Edition, Eindhoven University of Technology, 2008.

[15] B. Weber, H. A. Reijers, S. Zugal, W. Wild, The declarative approach to business process execution: An empirical test, in: P. van Eck, J. Gordijn, R. Wieringa (Eds.), Advanced Information Systems Engineering, Springer Berlin Heidelberg, Berlin, Heidelberg, 2009, pp. 470–485.

[16] S. Zugal, J. Pinggera, B. Weber, Toward enhanced life-cycle support for declarative processes, Journal of Software: Evolution and Process 24 (3) (2012) 285–302. `doi:10.1002/smr.554`.
URL `http://doi.wiley.com/10.1002/smr.554`

[17] T. R. Green, Cognitive dimensions of notations, People and computers V (1989) 443–460.

[18] T. R. G. Green, M. Petre, Usability analysis of visual programming environments: a 'cognitive dimensions' framework, Journal of Visual Languages & Computing 7 (2) (1996) 131–174.

[19] F. Chen, J. Zhou, Y. Wang, K. Yu, S. Z. Arshad, A. Khawaji, D. Conway, Robust multimodal cognitive load measurement, Springer, 2016.

[20] G. A. Miller, The magical number seven, plus or minus two: Some limits on our capacity for processing information., Psychological review 63 (2) (1956) 81.

[21] H. A. López, S. Debois, T. T. Hildebrandt, M. Marquard, The process highlighter: From texts to declarative processes and back., in: BPM (Dissertation/Demos/Industry), 2018, pp. 66–70.

[22] S. Debois, T. Hildebrandt, M. Marquard, T. Slaats, Hybrid Process Technologies in the Financial Sector, Proceedings of the Industry Track at the 13th International Conference on Business Process Management 2015 4 (2) (2015) 107–119.

[23] S. Zugal, J. Pinggera, B. Weber, The impact of testcases on the maintainability of declarative process models, in: T. Halpin, S. Nurcan, J. Krogstie, P. Soffer, E. Proper, R. Schmidt, I. Bider (Eds.), Enterprise, Business-Process and Information Systems Modeling, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 163–177.

[24] T. Slaats, D. M. M. Schunselaar, F. M. Maggi, H. A. Reijers, The Semantics of Hybrid Process Models, 2016, pp. 531–551. `doi:10.1007/978-3-319-48472-3_32`.
URL `http://link.springer.com/10.1007/978-3-319-48472-3_32`

[25] S. W. Sadiq, M. E. Orlowska, W. Sadiq, Specification and validation of process constraints for flexible workflows, Information Systems 30 (5) (2005) 349–378. `doi:10.1016/j.is.2004.05.002`.
URL `http://linkinghub.elsevier.com/retrieve/pii/S0306437904000456`

[26] S. Steinau, A. Marrella, K. Andrews, F. Leotta, M. Mecella, M. Reichert, Dalec: a framework for the systematic evaluation of data-centric approaches to process management software, Software & Systems Modeling 18 (4) (2019) 2679–2716. `doi:10.1007/s10270-018-0695-0`.
URL `https://doi.org/10.1007/s10270-018-0695-0`

[27] M. Weske, Business process management–concepts, languages, architectures, verlag (2007).

[28] P. Soffer, M. Kaner, Y. Wand, Towards understanding the process of process modeling: Theoretical and empirical considerations, Lecture Notes in Business Information Processing: Business Process Management Workshops - BPM 2011 99 (2012) 357–369. `doi:10.1007/978-3-642-28108-2_35`.

[29] S. Zugal, Applying Cognitive Psychology for Improving the Creation, Understanding and Maintenance of Business Process Models.

[30] C. W. Morris, Foundations of the theory of signs, International encyclopedia of unified science, University of Chicago Press, 1938.

[31] V. Fromkin, R. Rodman, N. Hyams, An Introduction to Language, Cengage Learning, 2013.

[32] G. Guizzardi, Ontological foundations for structural conceptual models, Vol. 015, 2005. `doi:10.1039/tf9625800697`.
URL `http://doc.utwente.nl/50826`

[33] T. Winograd, Frame representations and the declarative/procedural controversy, in: Representation and understanding, Elsevier, 1975, pp. 185–210.

[34] J. W. Lloyd, Practical advantages of declarative programming., in: M. Alpente, R. Barbuti, I. Ramos (Eds.), GULP-PRODE '94, 1994 Joint Conference on Declarative Programming, Proceedings, Vol. 1, 1994, pp. 3–17.

[35] P. Van-Roy, S. Haridi, Concepts, techniques, and models of computer programming, MIT press, 2004.

[36] D. Fahland, J. Mendling, H. A. Reijers, B. Weber, M. Weidlich, S. Zugal, Declarative versus Imperative Process Modeling Languages: The Issue of Maintainability, in: S. Rinderle-Ma, S. Sadiq, F. Leymann (Eds.), Business Process Management Workshops, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 477–488.

[37] S. Sadiq, W. Sadiq, M. Orlowska, Pockets of Flexibility in Workflow Specification, 2001, pp. 513–526. `doi:10.1007/3-540-45581-7_38`.
URL `http://link.springer.com/10.1007/3-540-45581-7_38`

[38] Object Management Group OMG, Case Management Model and Notation Version 1.0.
URL `http://www.omg.org/spec/CMMN/1.0/PDF`

[39] G. J. Nalepa, A. Lig\keza, K. Kaczor, Overview of Knowledge Formalization with XTT2 Rules, in: N. Bassiliades, G. Governatori, A. Paschke (Eds.), Rule-Based Reasoning, Programming, and Applications, Springer Berlin Heidelberg, Berlin, Heidelberg, 2011, pp. 329–336.

[40] W. van der Aalst, A. ter Hofstede, YAWL: yet another workflow language, Information Systems 30 (4) (2005) 245–275. `doi:10.1016/j.is.2004.02.002`.
URL `http://linkinghub.elsevier.com/retrieve/pii/S0306437904000304`

[41] G. Keller, M. Nüttgens, A. W. Scheer, Semantische Prozessmodellierung auf der Grundlage "ereignisgesteuerter Prozessketten (EPK)", Institut für Wirtschaftsinformatik Saarbrücken: Veröffentlichungen des Instituts für Wirtschaftsinformatik, Inst. für Wirtschaftsinformatik, 1992.

[42] O. Management Group, An OMG ® Semantics of Business Vocabulary and Business Rules TM Publication-Content-Model-for-SBVR.xml (May).
URL `https://www.omg.org/spec/SBVR/About-SBVR/{#}specification-metadata`

[43] A. Abbad Andaloussi, A. Burattin, T. Slaats, A. C. M. Petersen, T. T. Hildebrandt, B. Weber, Exploring the understandability of a hybrid process design artifact based on dcr graphs, in: I. Reinhartz-Berger, J. Zdravkovic, J. Gulden, R. Schmidt (Eds.), Enterprise, Business-Process and Information Systems Modeling, Springer International Publishing, Cham, 2019, pp. 69–84.

[44] A. A. Andaloussi, T. Slaats, A. Burattin, T. Thomas, B. Weber, Evaluating The Understandability of Hybrid Process Model Representations Using Eye Tracking : First Insights 1–6.

[45] J. De Smedt, J. De Weerdt, E. Serral, J. Vanthienen, Discovering hidden dependencies in constraint-based declarative process models for improving understandability, Information Systems 74 (2018) 40–52. `doi:10.1016/j.is.2018.01.001`.
URL `https://linkinghub.elsevier.com/retrieve/pii/S0306437916304732`

[46] W. Wang, M. Indulska, S. Sadiq, B. Weber, Effect of Linked Rules on Business Process Model Understanding, 2017, pp. 200–215. `doi:10.1007/978-3-319-65000-5_12`.
URL `http://link.springer.com/10.1007/978-3-319-65000-5_12`

[47] J. Recker, N. Safrudin, M. Rosemann, How Novices Model Business Processes, in: R. Hull, J. Mendling, S. Tai (Eds.), Business Process Management, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 29–44.

[48] A. Abbad Andaloussi, J. Buch-Lorentsen, H. A. López, T. Slaats, B. Weber, Exploring the Modeling of Declarative Processes Using a Hybrid Approach (2019) 1–15.

[49] W. M. Van Der Aalst, Process mining: Discovery, Conformance, and Enhancement of Business processes, Springer International Publishing, 2010.

[50] M. Westergaard, T. Slaats, Mixing Paradigms for More Comprehensible Models, 2013, pp. 283–290. `doi:10.1007/978-3-642-40176-3_24`.
URL `http://link.springer.com/10.1007/978-3-642-40176-3_24`

[51] G. De Giacomo, M. Dumas, F. M. Maggi, M. Montali, Declarative Process Modeling in BPMN, 2015, pp. 84–100. `doi:10.1007/978-3-319-19069-3_6`.
URL `http://link.springer.com/10.1007/978-3-319-19069-3_6`

[52] J. De Smedt, J. De Weerdt, J. Vanthienen, G. Poels, Mixed-paradigm process modeling with intertwined state spaces, Business and Information Systems Engineering 58 (1) (2016) 19–29. `doi:10.1007/s12599-015-0416-y`.

[53] U. Frank, Multi-perspective enterprise modeling: foundational concepts, prospects and future research challenges, Software & Systems Modeling 13 (3) (2014) 941–962. `doi:10.1007/s10270-012-0273-9`.
URL `https://doi.org/10.1007/s10270-012-0273-9`

[54] F. Dengler, D. Vrandečić, Wiki-Based Maturing of Process Descriptions, 2011, pp. 313–328. `doi:10.1007/978-3-642-23059-2_24`.
URL `http://link.springer.com/10.1007/978-3-642-23059-2_24`

[55] B. Kitchenham, Guidelines for performing Systematic Literature Reviews in Software Engineering 33 (Keele, UK, Keele University).

[56] D. Budgen, P. Brereton, Performing systematic literature reviews in software engineering, Proceeding of the 28th international conference on Software engineering - ICSE '06 (2006) 1051`arXiv:1304.1186`, `doi:10.1145/1134285.1134500`.
URL `http://portal.acm.org/citation.cfm?doid=1134285.1134500`

[57] J. Webster, R. T. Watson, Analyzing the Past to Prepare for the Future: Writing a Literature Review., MIS Quarterly 26 (2) (2002) xiii – xxiii. `arXiv:02767783`, `doi:10.1.1.104.6570`.

[58] R. Lu, S. Sadiq, G. Governatori, On managing business processes variants, Data and Knowledge Engineering 68 (7) (2009) 642–664. `doi:10.1016/j.datak.2009.02.009`.
URL `http://dx.doi.org/10.1016/j.datak.2009.02.009`

[59] B. Zimmermann, M. Doehring, Patterns for flexible BPMN workflows, Proceedings of the 16th European Conference on Pattern Languages of Programs - EuroPLoP '11 (2011) 1–9`doi:10.1145/2396716.2396723`.
URL `http://dl.acm.org/citation.cfm?doid=2396716.2396723`

[60] K. Hinkelmann, Business process flexibility and decision-aware modeling—the knowledge work designer, in: D. Karagiannis, H. C. Mayr, J. Mylopoulos (Eds.), Domain-Specific Conceptual Modeling: Concepts, Methods and Tools, Springer International Publishing, Cham, 2016, pp. 397–414. `doi:10.1007/978-3-319-39417-6_18`.
URL `https://doi.org/10.1007/978-3-319-39417-6_18`

[61] D. M. M. Schunselaar, T. Slaats, F. M. Maggi, H. A. Reijers, W. M. P. van der Aalst, Mining Hybrid Business Process Models: A Quest for Better Precision, in: Introducing Information Management, Vol. 1, Springer International Publishing, 2018, pp. 190–205. `doi:10.1007/978-3-319-93931-5_14`.
URL `http://link.springer.com/10.1007/978-3-319-93931-5_14`

[62] J. De Smedt, J. De Weerdt, J. Vanthienen, Fusion Miner: Process discovery for mixed-paradigm models, Decision Support Systems 77 (2015) 123–136. `doi:10.1016/j.dss.2015.06.002`.
URL `https://linkinghub.elsevier.com/retrieve/pii/S0167923615001165`

[63] J. De Smedt, J. De Weerdt, J. Vanthienen, Multi-Paradigm Process Mining: Retrieving Better Models by Combining Rules and Sequences, Ssrn`doi:10.2139/ssrn.2464066`.

[64] C. Ayora, V. Torres, B. Weber, M. Reichert, V. Pelechano, Vivace: A framework for the systematic evaluation of variability support in process-aware information systems, Information and Software Technology 57 (2015) 248–276.

[65] M. Milanovic, D. Gasevic, L. Rocha, Modeling Flexible Business Processes with Business Rule Patterns, in: 2011 IEEE 15th International Enterprise Distributed Object Computing Conference, IEEE, 2011, pp. 65–74. `doi:10.1109/EDOC.2011.25`.
URL `http://ieeexplore.ieee.org/document/6037561/`

[66] T. van Eijndhoven, M.-E. Iacob, M. L. Ponisio, Achieving Business Process Flexibility with Business Rules, in: 2008 12th International IEEE Enterprise Distributed Object Computing Conference, IEEE, 2008, pp. 95–104. `doi:10.1109/EDOC.2008.23`.
URL `http://ieeexplore.ieee.org/document/4634761/`

[67] J. De Smedt, J. De Weerdt, E. Serral, J. Vanthienen, Improving Understandability of Declarative Process Models by Revealing Hidden Dependencies, in: S. Nurcan, P. Soffer, M. Bajec, J. Eder (Eds.), Advanced Information Systems Engineering, Springer International Publishing, Cham, 2016, pp. 83–98.

[68] R. Cheng, S. Sadiq, M. Indulska, Framework for Business Process and Rule Integration: A Case of BPMN and SBVR, 2011, pp. 13–24. `doi:10.1007/978-3-642-21863-7_2`.
URL `http://link.springer.com/10.1007/978-3-642-21863-7_2`

[69] S. Zugal, J. Pinggera, B. Weber, Creating Declarative Process Models Using Test Driven Modeling Suite, in: S. Nurcan (Ed.), IS Olympics: Information Systems in a Diverse World, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 16–32.

[70] M. Marquard, M. Shahzad, T. Slaats, Web-Based Modelling and Collaborative Simulation of Declarative Processes, in: H. R. Motahari-Nezhad, J. Recker, M. Weidlich (Eds.), Business Process Management, Springer International Publishing, Cham, 2016, pp. 209–225.

[71] T. Graml, R. Bracht, M. Spies, Patterns of business rules to enable agile business processes, in: 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007), 2007, pp. 365–365. `doi:10.1109/EDOC.2007.35`.

[72] B. Sapkota, M. v. Sinderen, Exploiting rules and processes for increasing flexibility in service composition, in: 2010 14th IEEE International Enterprise Distributed Object Computing Conference Workshops, 2010, pp. 177–185. `doi:10.1109/EDOCW.2010.44`.

[73] K. Kluza, K. Kaczor, G. J. Nalepa, Enriching Business Processes with Rules Using the Oryx BPMN Editor, in: L. Rutkowski, M. Korytkowski, R. Scherer Rafałand Tadeusiewicz, L. A. Zadeh, J. M. Zurada (Eds.), Artificial Intelligence and Soft Computing, Springer Berlin Heidelberg, Berlin, Heidelberg, 2012, pp. 573–581.

[74] M. Milanovic, D. Gasevic, Towards a language for rule-enhanced business process modeling, in: 2009 IEEE International Enterprise Distributed Object Computing Conference, 2009, pp. 64–73. `doi:10.1109/EDOC.2009.12`.

[75] W. M. P. van der Aalst, M. Adams, A. H. M. ter Hofstede, M. Pesic, H. Schonenberg, Flexibility as a Service, 2009, pp. 319–333. `doi:10.1007/978-3-642-04205-8_27`.
URL `http://link.springer.com/10.1007/978-3-642-04205-8_27`

[76] P. Mangan, S. Sadiq, On Building Workflow Models for Flexible Processes, Journal Australian Computer Science Communications 24 (2) (2002) 103–109. `doi:http://doi.acm.org/10.1145/563932.563918`.

[77] P. J. Mangan, S. Sadiq, A constraint specification approach to building flexible workflows, Journal of Research and Practice in Information Technology 35 (1) (2003) 21–39.

[78] E. Mickevičiūtė, R. Butleris, Towards the Combination of BPMN Process Models with SBVR Business Vocabularies and Rules, 2013, pp. 114–121. `doi:10.1007/978-3-642-41947-8_11`.
URL `http://link.springer.com/10.1007/978-3-642-41947-8_11`

[79] K. Kluza, G. J. Nalepa, Formal Model of Business Processes Integrated with Business Rules, Information Systems Frontiers`doi:10.1007/s10796-018-9826-y`.
URL `http://link.springer.com/10.1007/s10796-018-9826-y`

[80] M. Scaife, Y. Rogers, External cognition: how do graphical representations work?, International Journal of Human-Computer Studies 45 (2) (1996) 185–213. `doi:10.1006/ijhc.1996.0048`.
URL `http://linkinghub.elsevier.com/retrieve/pii/S1071581996900488`

[81] M. Pesic, H. Schonenberg, W. van der Aalst, The Declare Service, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 327–343. `doi:10.1007/978-3-642-03121-2_12`.
URL `https://doi.org/10.1007/978-3-642-03121-2_12`

[82] M. Westergaard, T. Slaats, CPN tools 4: A process modeling tool combining declarative and imperative paradigms, CEUR Workshop Proceedings 1021.

[83] S. Zugal, C. Haisjackl, J. Pinggera, B. Weber, Empirical evaluation of test driven modeling, Int. J. Inf. Syst. Model. Des. 4 (2) (2013) 23–43. `doi:10.4018/jismd.2013040102`.
URL `http://dx.doi.org/10.4018/jismd.2013040102`

[84] I. Seeber, R. Maier, B. Weber, CoPrA: A Process Analysis Technique to Investigate Collaboration in Groups, in: 2012 45th Hawaii International Conference on System Sciences, IEEE, 2012, pp. 363–372. `doi:10.1109/HICSS.2012.184`.
URL `http://ieeexplore.ieee.org/document/6148651/`

[85] J. Pinggera, S. Zugal, B. Weber, Investigating the process of process modeling with cheetah experimental platform- Tool paper, CEUR Workshop Proceedings 603 (2010) 13–18.

[86] K. Holmqvist, M. Nyström, R. Andersson, R. Dewhurst, H. Jarodzka, J. van de Weijer, Eye Tracking: A comprehensive guide to methods and measures, OUP Oxford, 2011.
URL `https://books.google.dk/books?id=5rIDPV1EoLUC`

[87] R. N. Meghanathan, C. van Leeuwen, A. R. Nikolaev, Fixation duration surpasses pupil size as a measure of memory load in free viewing, Frontiers in Human Neuroscience 8. `doi:10.3389/fnhum.2014.01063`.
URL `http://journal.frontiersin.org/article/10.3389/fnhum.2014.01063/abstract`

[88] A. Poole, L. J. Ball, Eye tracking in human-computer interaction and usability research: Current status and future, in: Encyclopedia of Human-Computer Interaction. Pennsylvania: Idea Group, Inc, 2005.

[89] J. Gulden, A. Burattin, A. A. Andaloussi, B. Weber, From analytical purposes to data visualizations: a decision process guided by a conceptual framework and eye tracking, Software & Systems Modeling`doi:10.1007/s10270-019-00742-z`.
URL `https://doi.org/10.1007/s10270-019-00742-z`

[90] M. Zur Muehlen, M. Indulska, Modeling languages for business processes and business rules: A representational analysis, Information systems 35 (4) (2010) 379–390.

[91] O. I. Lindland, G. Sindre, A. Solvberg, Understanding quality in conceptual modeling, IEEE software 11 (2) (1994) 42–49.

[92] K. Figl, J. Mendling, M. Strembeck, J. Recker, On the cognitive effectiveness of routing symbols in process modeling languages, in: W. Abramowicz, R. Tolksdorf (Eds.), Business Information Systems, Springer Berlin Heidelberg, Berlin, Heidelberg, 2010, pp. 230–241.

[93] D. Moody, The "physics" of notations: Toward a scientific basis for constructing visual notations in software engineering, IEEE Transactions on Software Engineering 35 (6) (2009) 756–779. `doi:10.1109/TSE.2009.67`.

[94] J. Mendling, H. A. Reijers, W. M. P. van der Aalst, Seven process modeling guidelines (7pmg), Inf. Softw. Technol. 52 (2) (2010) 127–136. `doi:10.1016/j.infsof.2009.08.004`.
URL `http://dx.doi.org/10.1016/j.infsof.2009.08.004`

[95] J. Krogstie, Model-based development and evolution of information systems: A Quality Approach, Springer Science & Business Media, 2012.

[96] A. Burattin, M. Kaiser, M. Neurauter, B. Weber, Learning process modeling phases from modeling interactions and eye tracking data, Data & Knowledge Engineering 121 (2019) 1–17.

[97] C. Haisjackl, I. Barba, S. Zugal, P. Soffer, I. Hadar, M. Reichert, J. Pinggera, B. Weber, Understanding declare models: strategies, pitfalls, empirical results, Software and Systems Modeling 15 (2) (2016) 325–352. `doi:10.1007/s10270-014-0435-z`.
URL `https://doi.org/10.1007/s10270-014-0435-z`

[98] C. Haisjackl, P. Soffer, S. Y. Lim, B. Weber, How do humans inspect BPMN models: an exploratory study, Software and Systems Modeling 17 (2) (2018) 655–673. `doi:10.1007/s10270-016-0563-8`.

URL https://doi.org/10.1007/s10270-016-0563-8

[99] K. Figl, Comprehension of procedural visual business process models, Business & Information Systems Engineering 59 (1) (2017) 41–67. doi:10.1007/s12599-016-0460-2.
URL https://doi.org/10.1007/s12599-016-0460-2

[100] F. Garcia, F. Ruiz, C. A. Visaggio, A proposal and empirical validation of metrics to evaluate the maintainability of software process models, in: 2006 IEEE Instrumentation and Measurement Technology Conference Proceedings, 2006, pp. 1093–1097. doi:10.1109/IMTC.2006.328377.

[101] J. Claes, I. Vanderfeesten, F. Gailly, P. Grefen, G. Poels, The structured process modeling theory (spmt) a cognitive view on why and how modelers benefit from structuring the process of process modeling, Information Systems Frontiers 17 (6) (2015) 1401–1425. doi:10.1007/s10796-015-9585-y.
URL https://doi.org/10.1007/s10796-015-9585-y

[102] A. Bojko, Using eye tracking to compare web page designs: A case study, J. Usability Studies 1 (3) (2006) 112–120.
URL http://dl.acm.org/citation.cfm?id=2835663.2835665

Chapter 8

Article 2: Exploring the Modeling of Declarative Processes Using a Hybrid Approach

Exploring the Modeling of Declarative Processes Using a Hybrid Approach

Amine Abbad Andaloussi[1], Jon Buch-Lorentsen[1], Hugo A. López[2,5], Tijs Slaats[3], and Barbara Weber[1,4] ⋆

[1] Software and Process Engineering, Technical University of Denmark
2800 Kgs. Lyngby, Denmark – `amab@dtu.dk`
[2] Department of Computer Science, IT University of Copenhagen, Denmark
[3] Department of Computer Science, University of Copenhagen, Denmark
[4] Institute of Computer Science, University of St. Gallen, Switzerland
[5] DCR Solutions A/S, Denmark

Abstract. Process modeling aims at providing an external representation of a business process in the shape of a process model. The complexity of the modeling language, the usability of the modeling tool, and the expertise of the modeler are among the key factors defining the difficulty of a modeling task. Following a qualitative analysis approach, this work explores a hybrid modeling technique enhanced with a tool (i.e., the Highlighter) to guide the transition from informal text-based process descriptions to formal declarative process models. The exploratory results suggest that this technique provides cognitive support to modelers and hint towards an enhanced quality of process models in terms of alignment, traceability of process requirements and availability of documentation. The outcome of this work shows a clear opportunity for future work and provides a framework for further empirical studies.

1 Introduction

A process model is a visual/graphical representation of the different components of a business process, as well as their interrelations. The full understanding of a process tends to be a joint construction between different process design artifacts (process artifacts for short), including the business process model. In this paper, we examine an approach used to relate textual process artifacts and business process models during the Process of Process Modeling (PPM for short). This process is regarded as a "design activity" where a modeler develops an internal representation of the business process and externalizes it through one or many process artifacts [3]. Throughout this process, three levels of cognitive load are induced. *(1)* Intrinsic load is associated with the complexity of the material being processed, while *(2)* extraneous load is rising from the unnecessary

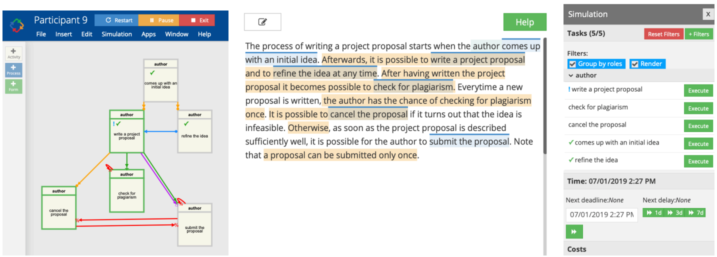
---

⋆ Work supported by the Innovation Fund Denmark project *EcoKnow* (7050-00034A), the Danish Council for Independent Research project *Hybrid Business Process Management Technologies* (DFF-6111-00337), and the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement BehAPI No.778233.

representational complexity of the task. *(3)* Germane load, in turn, is associated with the effort invested in building an appropriate scheme to organize new information efficiently [5]. During a modeling session, intrinsic load emerges from the complexity of inferring a mental model from a set of process specifications. Extraneous load raises from the formulation of the textual process description and the complexity of the modeling tool. While intrinsic load is inherent to the task and thus unavoidable, efforts can be made to reduce the extraneous load by improving the quality of the tool-support and enhancing the PPM experience.

When considering the declarative modeling paradigm, the requirement for lowering extraneous load in favor of extra intrinsic processing becomes more stringent. This is due to the understandability of declarative languages, which is shown to be controversial especially for novice end-users [8]. A hybrid modeling approach can, in turn, be used to facilitate the modeling of declarative business processes and provide additional channels to support the PPM through a set of interrelated process artifacts. In this vein, the Highlighter [11] was introduced.

The Highlighter (cf. Fig. 1b) is integrated with the default Dynamic Condition Response (DCR [10]) graphical modeling tool (shortly, the Modeler, cf. Fig. 1a) and a guided simulation (cf. Fig. 1c). The tool displays a process model and an annotatable textual description side-by-side allowing to map the specifications in the textual process description with the corresponding model elements (i.e., activities, roles and relations). During a typical modeling session, end-users can design process models by highlighting activities, roles and relations in the process description, then intertwine with the Modeler and the guided simulation to reconcile and validate the process model. Following a qualitative research approach, this work aims at exploring the understandability of such a hybrid process artifact. The remainder of this paper is organized as follows. Sect. 2 provides an overview of the existing hybrid process artifacts. Sect. 3 explains the research method. Sect. 4 reports the obtained findings. Sect. 5 provides a discussion, while Sect. 6 wraps up the key findings and presents future work.



The process of writing a project proposal starts when the author comes up with an initial idea. Afterwards, it is possible to write a project proposal and to refine the idea at any time. After having written the project proposal it becomes possible to check for plagiarism. Everytime a new proposal is written, the author has the chance of checking for plagiarism once. It is possible to cancel the proposal if it turns out that the idea is infeasible. Otherwise, as soon as the project proposal is described sufficiently well, it is possible for the author to submit the proposal. Note that a proposal can be submitted only once.

Tasks (5/5)
Filters: ☑ Group by roles ☑ Render
author
! write a project proposal
check for plagiarism
cancel the proposal
✓ comes up with an initial idea
✓ refine the idea
Time: 07/01/2019 2:27 PM
Next deadline:*None* Next delay:*None*
07/01/2019 2:27 PM

(a) DCR Modeler (b) Highlighter (c) Simulator

Fig. 1: A hybrid process artifact combining the Modeler, the Highlighter and the simulation tools. Available online as part of the DCR platform at `https://dcrgraphs.net/`

2 Background and Related Work

Hybrid process representations are introduced in the literature in two contexts: *(a)* to designate hybrid languages (e.g., [14]) or *(b)* to describe hybrid process artifacts. While hybrid languages combine existing languages to enable a concise and precise representation of business processes, hybrid process artifacts combine two or more process artifacts overlapping in the description of some aspects of the business process [1]. The emergence of hybrid process artifacts is driven by three main motivations: *(1)* supporting the understandability of process models (cf., [1,2]), *(2)* enhancing the maintainability of process models (cf., [16]), and *(3)* improving the modeling of business processes.

Similar to this work, Dengler and Denny, in [7], propose a hybrid process artifact that combines process models and textual descriptions, embedded in a wiki-based platform. The proposed representation aims at improving the PPM experience by enabling different stakeholders to extract process knowledge and to express business processes using both formal and informal constructs. The findings of a qualitative analysis show that the proposed approach supports better knowledge elicitation. With the same idea in mind, Pinggera et al. in [12] propose the Literate Process Modeling (LiProMo) approach aiming at interweaving annotations and graphical process models to enhance the communication when modeling business processes.

3 Research Method

This section introduces the research questions, presents the subjects who took part in this study, describes the material and the procedure followed to run the study and explains the approach used to analyze the collected data.

Research Questions: The Highlighter aims at enhancing the PPM experience by providing a tool-support allowing to facilitate the transition from a textual process description to a graphical process model. In order to investigate this support, it is necessary to understand the way the Highlighter is used in practice. To this end, the first research question is formulated as follows: **RQ1: How do users engage with a modeling task using the Process Highlighter?**

By enhancing the PPM experience, the Highlighter is expected to positively affect the perceived quality of the produced models. To explore this angle, the second research question is formulated as follows: **RQ2: In what aspects can the Highlighter help to improve the quality of process models?**

Participants The participants who took part in this study included novice subjects from industrial and education environments. In the former, 7 employees from the Syddjurs municipality in Denmark, and from the latter, 10 students from the Technical University of Denmark (DTU).

Material The material used to conduct this study originates from a process introduced by Reichert and Weber in [13][p. 349]. This process describes the writing of a project proposal. The material was presented in Danish at Syddjurs municipality and in English at DTU. A copy of the material is available online at `http://andaloussi.org/papers/ER2019/material.pdf`

Procedure The study was conducted in both Syddjurs municipality and DTU. Participants were introduced to the modeling notation and the use of the Highlighter in both locations. Then, participants were given a familiarization task on PPM using the tool and the notation. Next, the participants were given the description of the *project proposal* process and were asked to use the Highlighter to derive the corresponding process model. We collected participant's insights about their experience with the tool from retrospective think-aloud sessions.

Analysis Approach In order to address our research questions, two different analyses have been performed. At the first stage, we have extracted the interactions of the users with the DCR modeling platform. This data were filtered to keep only the interactions associated with adding activities, roles and relations. Next, these interactions were split between those using the Highlighter, and those using the Modeler. During the analysis, the interactions were aggregated over all the modeling sessions and projected according to their time-occurrence into a rhythm eye chart [9]. An example of such a visualization is shown in Fig. 2. The ring structure represents a time-line, the different percentages refer to the progress in relative time. Events (i.e., interactions) are projected as thin lines onto the ring and events of similar type (e.g., interaction with the Highlighter) are depicted with the same color. Besides the user interactions, the collected verbal data were transcribed and analyzed following a qualitative coding approach based on concepts from grounded theory [6].

4 Findings

This section reports the findings. Sect. 4.1 scrutinizes the way users engage with a modeling task using the Highlighter. Sect. 4.2 explores whether the proposed hybrid modeling approach can improve the quality of process models.

4.1 How do Users Engage With a Modeling Task Using the Process Highlighter? (RQ1)

The users' interactions collected throughout the modeling sessions provide deepened insights into the way end-users engaged with the Highlighter. As shown in Fig. 2, most of the interactions with the Highlighter occurred during the first quarter of the modeling session, which in turn, suggests that most end-users initiated the modeling using the Highlighter and then progressively moved to the Modeler. To further substantiate this modeling pattern, the users' interactions were scrutinized to identify the common interactions within each of the process artifacts. As shown in Fig. 3a, a larger portion of activities were appended to the model using the Highlighter. Similarly, Fig. 3b shows that most roles were added using the Highlighter. Unlike activities and roles, Fig. 3c shows that relations were mostly added using the Modeler, which in turn suggests that the

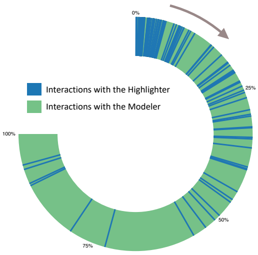
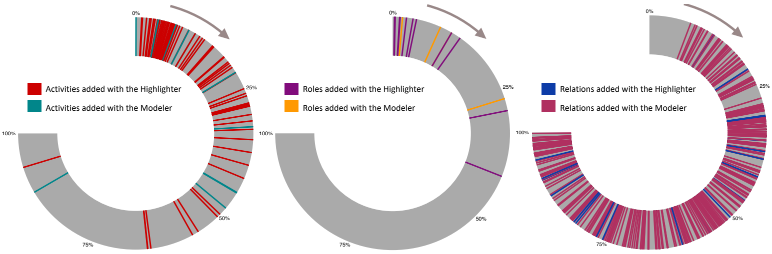


Fig. 2: The interactions associated with the Highlighter and the Modeler tool.



(a) Adding activities (b) Adding roles (c) Adding relations

Fig. 3: Interactions associated with adding activities, roles and relations.

Highlighter was not extensively used to add relations. These users' interactions come in line with the subjective insights provided by the participants during the think-aloud. Indeed, most participants affirmed using the Highlighter to identify activities and roles from the process description and resort to the Modeler to add relations. These insights raise the following questions: *(1) Why is the Highlighter perceived more efficient to identify and add activities and roles? (2) What makes the use of the Modeler tool more attractive for adding relations to the model?*

To answer both questions, we turn to the qualitative coding of the verbal data. In respect to *(1)*, the participants mentioned that the tool provides a kick-start to process modeling and helps in developing an overview of the business process (e.g., *"Definitely, I think it is way easier to use the Highlighter to create the activities and it gives a better overview"*). Moreover, some participants have associated the use of the Highlighter with its ability to provide structure and to decompose the complexity of the process description (e.g., *"it is* [referring to the Highlighter] *a nice way to structure the text"*). Other participants mentioned that the Highlighter can help to memorize the process specifications and to draw attention to specific fragments of the process description (e.g., *"It was faster that was the main focus. at least I feel that* [it] *helps speed things up. I did not really notice that text was highlighted because I already knew what I had highlighted myself, so I mainly focused on the relations that could be between them"*).

In respect to *(2)*, while the identification of activities and roles was straightforward for most participants, many of them faced difficulty when trying to add relations in the Highlighter. Some participants justified their abstention with the argument that the Modeler tool provides a two-dimensional visualization allowing to perceive the interplay between the different activities (e.g., *"It just seemed easier once the visual aspect of the activities were done, then you could just connect them directly"*). In addition, some participants struggled to locate the exact textual fragment referring explicitly to a specific constraint in the process description. This struggle might be due to the phrasing of the process description (e.g., *"For the relations, I'm not sure it's the problem of the Highlighter or on the formulation of the text"*). Unlike activities and roles which are often explicit in the process description, relations may not be always explicit in the text.

4.2 In What Aspects Can the Highlighter Help to Improve the Quality of Process Models? (RQ2)

From the think-aloud, it has emerged that the mapping between the process model and the process description supports better traceability of the process specification (e.g., *"Using the Highlighter makes sense in the sense that it adds traceability . . . it helped me map the relations to the requirements"*) and enables a wider coverage of the requirements in the process description (e.g., *"It would be useful after and it is also useful during because I can see whether I already covered some piece of text"*). In addition, the participants' quotes indicate that the Highlighter was used to check the alignment between the process description and the process model (e.g., *"It* [referring to the Highlighter] *becomes indispensable as a method to verify whether the process fits with what has been described"*[6]). Last but not least, some participants emphasized the importance of using the Highlighter as a mans to document their process models (e.g., *"I think it is very useful as a documentation tool and documentation can also be very useful during the process"*). Indeed, the explicit links between the process model and the textual process description can serve for documenting the semantics of the model and enabling modelers to justify their modeling choices [12].

5 Discussion

The findings of this exploratory study provide several indications about the perceived benefits of the Highlighter. Both the subjective insights obtained from the participants and the user interactions extracted from the modeling platform show that the Highlighter was perceived more efficient to identify and append activities and roles to the model. These insights fall in line with the conclusions drawn from cognitive psychology. Indeed the use of the Highlighter to mark-up specific fragments of the process description (e.g., activities, roles) can be associated with a well-known phenomenon referred in cognitive psychology as the *isolation effect* [15]. This effect is shown to increase the reader attention on

---

[6] Quote translated from Danish

specific parts of the text and help memorizing them [4] [15]. This, in turn, can potentially explain the participants' insights related to the increased memory and attention when using the Highlighter and to some extent support the other insights about the ability of the Highlighter to provide overview and structure as well as to reduce the complexity of the process description (cf. Section 4.1). In addition to that, the quotes of several participants indicate that the Highlighter can support increased traceability, enhanced coverage and better alignment between the process model and the corresponding process description. However, when it gets to identify relations in the model, the Highlighter was challenging. As mentioned in Section 4.1, This challenge is associated with the difficulty in identifying the right text reflecting a certain constraint in the process model, which can be due to the phrasing of the process description.

All these insights provide indications about the extraneous load arising from using the tool. Indeed, the cognitive support provided by the Highlighter can reduce the complexity of the modeling task and contribute to an enhanced PPM experience. However, the implicitness of some constraints in the process description can add an extra layer of complexity when trying to map them to DCR relations, which in turn can induce a higher extraneous load. Hence, the use of the Highlighter can be presumably more effective with process descriptions comprising explicit constraints.

Finally, it has to be noted that the outcome of this exploratory work can be subject to limitations mainly with regards to the number of participants who participated in the study. Therefore, it is hard to generalize the reported findings and draw strong conclusions about the use of hybrid process artifacts in general and the Highlighter in particular. Nevertheless, the outcome of this work provides interesting insights emerging from the users' experience and sheds light on the direction of subsequent empirical investigations.

6 Conclusion and Future Work

This work summarizes the findings of an exploratory study investigating the modeling of DCR graphs with the support of the Highlighter. The results suggest that the use of the Highlighter is associated with increased support in PPM and hints toward an enhanced quality of process models. The outcome of this study provides strong indications for the direction of future work. Based on the conclusions drawn from cognitive psychology, we hypothesize that *(a)* the Highlighter reduces the cognitive load induced during a modeling task. Moreover, following the insights about the explicit mapping between the process specifications and the corresponding model elements we hypothesize that *(b)* the Highlighter improves model comprehension and clarifies the semantics of the model. Concerning the quality of process models, we hypothesize that *(c)* the Highlighter provides better alignment between the process description and the process model and enables covering the majority of the requirements mentioned in the text.

These hypotheses define our direction for future work. Following a quantitative analysis approach, we are planning a series of experiments to test and validate each of these hypotheses in controlled experimental settings. Moreover, it would be worth to investigate in the up-coming studies the support offered by the Highlighter when integrated with other process modeling languages from both the declarative and the imperative paradigms. The findings will serve as a basis to validate the usability of the Highlighter and will help to improve the design of similar hybrid process artifacts.

References

1. A. Abbad Andaloussi, A. Burattin, T. Slaats, C. A. Petersen, T. T. Hildebrandt, and B. Weber. Exploring the Understandability of a Hybrid Process Design Artifact Based on DCR Graphs. In *BPMDS*, LNBIP, pages 1–15. Springer, 2019.
2. A. Abbad Andaloussi, T. Slaats, A. Burattin, T. T. Hildebrandt, and B. Weber. Evaluating the understandability of hybrid process model representations using eye tracking: First insights. In *BPM Workshops*, pages 475–481. Springer, 2019.
3. A. Burattin, P. Soffer, D. Fahland, J. Mendling, H. A. Reijers, I. Vanderfeesten, M. Weidlich, and B. Weber. Who is behind the model? classifying modelers based on pragmatic model features. In *BPM*, pages 322–338, 2018.
4. V. M. Cashen and K. L. Leicht. Role of the isolation effect in a formal educational setting. *Journal of Educational Psychology*, 61(6p1):484, 1970.
5. F. Chen, J. Zhou, Y. Wang, K. Yu, S. Z. Arshad, A. Khawaji, and D. Conway. *Robust Multimodal Cognitive Load Measurement*, pages 13–32. Springer, 2016.
6. J. Corbin and A. Strauss. *Basics of Qualitative Research: Techniques and Procedures for Developing Grounded Theory.* SAGE Publications, 2014.
7. F. Dengler and D. Vrandecic. Wiki-based maturing of process descriptions. In *BPM*, volume 6896 of *LNCS*, pages 313–328. Springer, 2011.
8. D. Fahland, D. Lübke, J. Mendling, H. Reijers, B. Weber, M. Weidlich, and S. Zugal. Declarative versus imperative process modeling languages: The issue of understandability. In *BPMDS/EMMSAD*, LNBIP, pages 353–366. Springer, 2009.
9. J. Gulden. Visually comparing process dynamics with rhythm-eye views. In *BPM Workshops*, LNBIP, pages 474–485. Springer, 2016.
10. T. T. Hildebrandt and R. R. Mukkamala. Declarative Event-Based Workflow as Distributed Dynamic Condition Response Graphs. *EPTCS*, 69:59–73, 2011.
11. H. A. López, S. Debois, T. T. Hildebrandt, and M. Marquard. The process highlighter: From texts to declarative processes and back. In *BPM (Dissertation/Demos/Industry)*, volume 2196 of *CEUR*, pages 66–70. CEUR-WS.org, 2018.
12. J. Pinggera, T. Porcham, S. Zugal, and B. Weber. LiProMo-Literate process modeling. In *CAiSE Forum*, volume 855 of *CEUR*, pages 163–170. CEUR-WS.org, 2012.
13. M. Reichert and B. Weber. *Enabling Flexibility in Process-Aware Information Systems.* Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
14. T. Slaats, D. M. M. Schunselaar, F. M. Maggi, and H. A. Reijers. The semantics of hybrid process models. In *OTM 2016 Conferences*, LNCS. Springer, 2016.
15. H. von Restorff. Über die wirkung von bereichsbildungen im spurenfeld. *Psychologische Forschung*, pages 299–342, 1933.
16. S. Zugal, J. Pinggera, and B. Weber. The impact of testcases on the maintainability of declarative process models. In *BPMDS/EMMSAD*, LNBIP, pages 163–177. Springer, 2011.

Chapter 9

Article 3: 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 manuscript No.
(will be inserted by the editor)

Exploring How Users Engage with Hybrid Process Artifacts Based on Declarative Process Models

A behavioral analysis based on eye-tracking and think-aloud

Amine Abbad Andaloussi · Francesca Zerbato · Andrea Burattin · Tijs Slaats · Thomas T. Hildebrandt · Barbara Weber

Received: date / Accepted: date

Abstract *Context:* Process design artifacts have been increasingly used to guide the modeling of business processes. To support users in designing and understanding process models, different process artifacts have been combined in several ways leading to the emergence of the so-called "hybrid process artifacts". While many hybrid artifacts have been proposed in the literature, little is known about how they can actually support users in practice.
Objective: To address this gap, this work investigates the way users engage with hybrid process artifacts during comprehension tasks. In particular, we focus on a hybrid representation of DCR graphs (DCR-HR) combining a process model, textual annotations and an interactive simulation.
Method: Following a qualitative approach, we conduct a multi-granular analysis exploiting process mining, eye-tracking techniques, and verbal data analysis to scrutinize the reading patterns and the strategies adopted by users when being confronted with DCR-HR.
Results: The findings of the coarse-grained analysis provide important insights about the behavior of domain experts and IT specialists and show how user's background and task type change the use of hybrid process artifacts. As for the fine-grained analysis, user's behavior was classified into goal-directed and exploratory and different strategies of using the interactive simulation were identified. In addition, a progressive switch from an exploratory behavior to a goal-directed behavior was observed. These insights pave the way for an improved development of hybrid process artifacts and delineate several directions for future work.

Keywords Process Models · Hybrid Process Artifacts · DCR graphs · Eye-Tracking · Think-aloud · Behavioral Analysis

1 Introduction

The design and development of Process-Aware Information Systems (PAIS) encompasses the creation of several process design artifacts (process artifacts, for short) aimed to support users in modeling, enacting and managing business processes. Such artifacts may include constructs, models, methods and instantiations [1], which are created to support users in solving a specific problem throughout the different phases of the business process life-cycle.

Over the years, process artifacts have been more and more integrated in the development of PAIS, leading to

---

Work supported by the Innovation Fund Denmark project *EcoKnow* (7050-00034A); the fourth author additionally by the Danish Council for Independent Research project *Hybrid Business Process Management Technologies* (DFF-6111-00337).

Amine Abbad Andaloussi
Technical University of Denmark, Denmark
E-mail: amab@dtu.dk

Francesca Zerbato
University of St. Gallen, Switzerland
E-mail: francesca.zerbato@unisg.ch

Andrea Burattin
Technical University of Denmark, Denmark
E-mail: andbur@dtu.dk

Tijs Slaats
University of Copenhagen, Denmark
E-mail: slaats@di.ku.dk

Thomas T. Hildebrandt
University of Copenhagen, Denmark
E-mail: hilde@di.ku.dk

Barbara Weber
University of St. Gallen, Switzerland
E-mail: barbara.weber@unisg.ch

hybrid solutions that loosely combine different artifacts with the aim to support the design and comprehension of process models [2,3], especially those lying under the umbrella of the declarative paradigm [4–9]. Such hybrid solutions combine (graphical) process models with textual process specifications [6,8] and interactive simulations [4,10].

In this work, we refer to *hybrid process artifacts* as representations combining two or more design artifacts (e.g., process models, textual annotations or interactive simulations) overlapping in the description of some business process aspects [11] and we specifically consider declarative process models [12].

In literature, there have been several hybrid process artifacts proposed to tackle the notorious limitations of declarative process models, particularly with regard to their understandability and maintainability [4,5,13,14]. Indeed, to support flexibility in process design, declarative process models capture constraints on the allowed activity flows. Hence, their interpretation requires a constant awareness of the states of all the constraints in the model throughout process execution, which can cause a high burden when being confronted with complex models [15]. This challenge also affects the maintainability of declarative process models. Herein, the support of a hybrid process artifact can ease the interpretation of the process model and enhance its maintainability.

While previous research has focused on proposing hybrid process artifacts to support users in designing and understanding process models [5,7,16], at this time, there is a lack of empirical research about how these hybrid representations are used. In particular, an in-depth understanding of how users engage with hybrid process artifacts is needed. Besides, an understanding of the benefits and challenges associated with the use of the different artifacts is required to contextualize the observed behaviors. Moreover, it is unclear whether users follow certain strategies when engaging with hybrid process artifacts.

To address this gap, the goal of this paper is to provide insights on how people engage with hybrid process artifacts. More specifically, this paper investigates how people use a specific hybrid artifact during comprehension tasks. In particular, we focus on the hybrid representation of DCR Graphs (DCR-HR, for short). This hybrid process artifact combines a declarative process model represented as a Dynamic Condition Response (DCR) graph [17], its textual specification and an interactive simulation. DCR Graphs are a well-known declarative process modeling language based on directed graphs whose nodes represent events and whose edges capture the relationship between them [17]. DCR Graphs benefit from the support of the DCR Graphs Portal [18], a research-based commercial tool supporting the design, enactment and analysis of DCR Graphs. The DCR Graphs Portal features a graphical web-based editor, a textual process specification that can be visually linked to parts of the process model and an interactive simulator that can be directly enabled from the editor and visualizes the process execution directly on the DCR Graph. Being widely adopted by industrial and governmental institutions in Denmark, the DCR Graphs Portal and, more in general, DCR Graphs are a valuable candidate for user behavior studies compared to solutions based on other declarative process modeling languages such as DECLARE [15] that have not been commercialized so far.

To gain understanding of how people engage with DCR-HR, we designed an exploratory study asking people to perform a set of comprehension tasks using DCR-HR. The goals of the study are to (i) observe the distribution of attention among the different artifacts considering different groups of stakeholders and different kinds of comprehension tasks, (ii) gather insights on the perceived benefits and challenges associated to each artifact, and (iii) identify common strategies describing how people approach comprehension tasks and use different artifacts over time.

For collecting data we rely on two different well-known approaches, namely eye-tracking [19] and retrospective think-aloud. Then, we favor a qualitative data analysis approach to explore user's behavior.

Eye-tracking has been applied to numerous fields in order to understand the complete user experience during the execution of different tasks [20], as it provides insights on the natural interaction of a user with a system [21]. In this paper, we collect eye-tracking data of users engaging with DCR-HR and analyze them qualitatively following two distinct approaches. A first coarse-grained analysis exploits process mining techniques [22] and attention maps [19] to investigate how the three different artifacts of DCR-HR are used, specifically focusing on how groups of stakeholders with different backgrounds relate to DCR-HR while performing different kinds of tasks. As our focus is on attention distribution, during this analysis phase we are interested in gaining insights into *how much* each artifact is used individually and in combination with others. Then, we conduct a more fine-grained analysis considering the different elements of the DCR Graph individually and exploring *how* artifacts are used *over time*. During this analysis, we look deeper into temporal patterns and observe common strategies describing how people engage with DCR-HR during the comprehension task. In this analysis, we rely on scarf-plot visualizations [23] and, then, build on the well-known dichotomy of goal-directed and exploratory search behaviors [24, 25] to categorize the identified strategies.

Retrospective think-aloud is a research method used to verbalize users' thoughts after the execution of a certain task [19]. In this paper, we use retrospective think-aloud to extract the subjective insights of the participants who took part in the exploratory study, focusing on perceived benefits and challenges associated with the artifacts of DCR-HR. This allows us to obtain explicit user feedback, which is needed for having a complete picture of *why* users behaved in a certain way and for enhancing the interpretation of eye-tracking data during the described analyses.

In general, using behavioral data and think-aloud can inform us on the use of hybrid process artifacts (through implicit and explicit feedback [26]). Overall, the results of the exploratory study suggest that different groups of stakeholders tend to use different artifacts of DCR-HR and that usage changes based on the type of task being executed. Indeed, the users' background seems to affect perceived benefits and challenges, thus influencing the way different artifacts are used for achieving a specific purpose. When examining the use of artifacts over time, we found that users follow different strategies to interact with the artifacts of DCR-HR and observed different ways in which the interactive simulation was used. In addition, we noticed that people tend to switch from an exploratory behavior to goal-directed behavior progressively. The outcomes of this study confirm that the use of different process artifacts enhances the experience of users with different backgrounds, also based on the kind of task being executed. Besides, our findings pave the path for future research in the direction of improving the design of hybrid process artifacts, for example by considering explicit user preferences and implicit feedback to add or eliminate certain artifact features.

This paper extends original work initially presented in [10] by providing a broader and more complete overview of how users engage with DCR-HR. In particular, we introduce a novel fine-grained analysis which considers DCR Graphs and centers around the temporal dimension of the eye-tracking data. By observing how the users' behavior unfolds over time, we are able to identify interesting strategies describing how users engage with DCR-HR.

The remainder of this paper is organized as follows. Section 2 provides the reader with useful background concepts. Section 3 introduces related work. Section 4 presents the research method followed to design the exploratory study. Section 5 reports the results of the analysis. Section 6 discusses the main findings and highlights the interesting outcomes of this research, as well as its limitations. Finally, Section 7 concludes the paper and delineates the directions for future work.

2 Background

This section presents the main notions employed throughout this paper. We start with a general introduction to hybrid process artifacts (Section 2.1), followed by a description of DCR-HR, (Section 2.2). Finally, we introduce eye-tracking as one of the core methodologies behind our study (Section 2.3).

2.1 Hybrid Process Artifacts

Hybrid process artifacts combine two or more process artifacts (e.g., process models, textual annotations or interactive simulations) overlapping in the description of some business process aspects [11]. Hybrid process artifacts have been proposed in the literature to address several challenges within the areas of process modeling (for a systematic literature review see [11]), in particular to address open challenges in the use of declarative languages [27,28] caused by their limited understandability [29] and maintainability [30]. The limited capacity of humans when dealing with constraints is among the key challenges in that respect. Indeed, a full understanding of a declarative process model requires being aware of the states of all the constraints in the model throughout the whole process execution [28]. This requirement gets more complicated when considering the implicit constraints (also called "hidden dependencies [5]) between the model activities and all the possible ways in which they could interact. As the capacity of the human working memory is usually limited to 7 ± 2 items [31], interpreting declarative process models with too many constraints, without the support of others process artifacts (e.g., interactive simulations or textual annotations), becomes a challenging task. The limitations of declarative languages go beyond their understandability, as their maintainability is also quickly hampered by hidden dependencies. Considering the entanglement of hidden dependencies and the abundance of ways in which they could interact, it becomes challenging to infer the set of constraints affected by a change of the process specifications and to ensure the consistency of the model after altering some of its constraints. Hence, without the support of additional artifacts, the maintainability of a declarative process model is prone to misalignment and non-compliance with the process specification.

2.2 DCR-HR: A Hybrid Representation of DCR Graphs

DCR-HR is a hybrid process artifact combining (i) a DCR Graph [17], with (ii) textual process specifications and (iii) a simulation allowing to evaluate the behavior of the process model. DCR Graphs and DCR-HR have been developed through a close collaboration between academia and industry [32,33], combining research into formal methods and declarative notations with the development of a commercial modelling tool[1] and its application to real-world cases [34].

The inclusion of textual process specifications in the presentation was driven by the ongoing EcoKnow research project[2], where the DCR technology is being applied to support the effective digitization of citizen processes. The use of DCR-HR, combining textual legal paragraphs with the graphical DCR notation and guided simulation, is used to empower knowledge workers in Danish municipalities by enabling them to make sense of digitized models of the law.

The version of DCR-HR considered within this paper is based on this application within the EcoKnow project and specifically refers to a process derived from section §45 of the Danish "Consolidation Act on Social Services" [3]. Its layout is depicted in Fig. 1. Here, we have on the left a DCR Graph modelling the aforementioned process and consisting of: (i) boxes denoting the activities of the process, e.g. *offer 15 hours of assistance* and *designate a person*, and (ii) arrows between the boxes denoting the constraint on the process, e.g the yellow arrow with a dot on the end between the former two activities. The activities can be assigned a role, placed in the bar above the box, e.g. *Receiver* for *designate a person.* They can also be nested inside each other, indicating a form of hierarchy, e.g. all activities are a part of *paragraph 45*. When constraints are drawn between such nestings they apply to all child activities. As is the norm for declarative notations, unconstrained activities can be executed freely, i.e., at any time and any number of times. DCR Graphs include five types of constraints: (i) the *condition*, drawn as a yellow arrow with a dot on the end, denotes that before one activity can be executed, another needs to have been done at least once in the past; (ii) the *response*, drawn as a blue arrow with a dot at the start, denotes that after one activity is executed, some other activity becomes required and needs to be done before the process can be finalized; (iii) the *exclusion*, drawn as a red arrow with a percentage sign at the end, denotes that when one activity is executed another activity is removed of the process; (iv) the *inclusion*, drawn as a green arrow with a plus sign at the end, denotes that when one activity is executed another activity is added back into the process; finally (v) the *milestone*, drawn as a purple arrow with a diamond at the end, denotes that while one activity is required to be done another activity is blocked from executing.

At the bottom of Fig. 1 we have the law text, taken directly from the relevant laws that govern the process. Each fragment of the law text is linked to the currently selected activity: by selecting *paragraph 45* users will be able to visualize the entire law text for the current process, while by selecting *designate a person* they will see only the part of the law that is relevant for this activity.

Finally, at the right side of Fig. 1, we have the interactive simulation. This consists of (i) a clickable list of currently executable tasks, (ii) a textual log of what has already been simulated, (iii) a swim-lane representation of this log. Each time an activity is executed for simulation, it will be added to the textual log and swim-lane, and the list of currently executable tasks will update.

The combination of these different artifacts is meant to allow users to form their own strategies when reading the model, based on their personal background and the type of comprehension tasks that they are trying to solve. For example, when asked a question about the law some users may use the graphical process model to find the answers, while others may focus on the text. For highly operational questions, some users may prefer to use the guided simulation, as it allows them to quickly try out different execution scenarios without having to memorize all the possible execution traces.

2.3 Eye-tracking

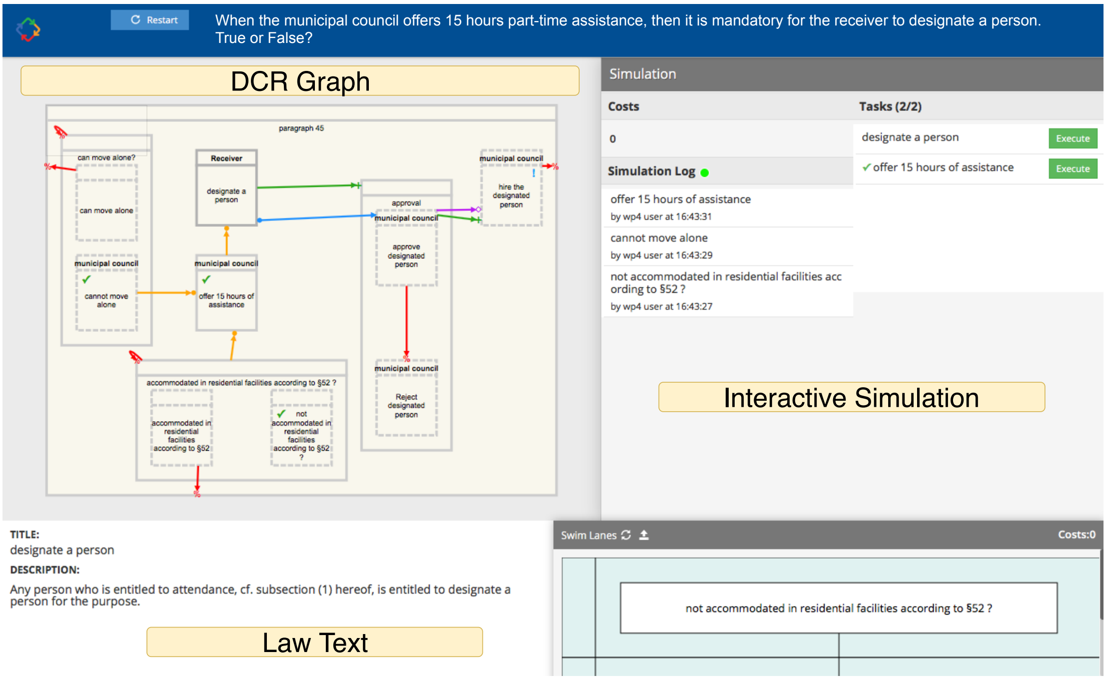
Eye-tracking is a widely adopted methodology allowing researchers to track humans' gaze interactions with external stimuli [20]. Eye movements are recorded by eye-tracking devices as gaze data, which in turn are used to derive a set of oculomotor events such as *fixations* and *saccades*. The detection of these events is associated with a set of properties such as duration, amplitude, and velocity [19]. A fixation refers to the time span when the eye remains still at a specific position of the stimulus [19]. An example of fixation is the time the eye stops at a word while reading a sentence. A saccade refers to the rapid eye movement occurring between fixations [19]. When reading a sentence, saccades occur when the reader moves from one word to another.

---

[1] `http://www.dcrgraphs.net`

[2] `https://ecoknow.org`

[3] `http://english.sm.dk/media/14900/consolidation-act-on-social-services.pdf`



When the municipal council offers 15 hours part-time assistance, then it is mandatory for the receiver to designate a person.
True or False?

TITLE:
designate a person

DESCRIPTION:
Any person who is entitled to attendance, cf. subsection (1) hereof, is entitled to designate a person for the purpose.

Simulation Log

offer 15 hours of assistance
by wp4 user at 16:43:31

cannot move alone
by wp4 user at 16:43:29

not accommodated in residential facilities according to §52 ?
by wp4 user at 16:43:27

Fig. 1 A view showing the DCR-HR layout. The hybrid process artifact comprises a DCR Graph, law text and an interactive simulation.

The availability of fixations and saccades allows for a wide range of statistical and visual analyses of human behavior. While statistical analyses (i.e., descriptive statistics, inferential statistics, and statistical modeling) provide the basis for hypothesis testing, their use requires aggregating data to a level where the application of statistical methods is possible. However, this comes at the cost of providing rather coarse and limited insights about the human visual behavior [35]. In addition, the interpretation of the metrics inferred from the aggregated data is tightly coupled with the context and the application where these metrics have evolved. As a result, the mapping between the numerical data and the real perceptual and cognitive features of the participants remains uncertain.

Alternatively, visual analyses follow an exploratory approach to investigate the way people approach a stimulus. Graphical plots allow visualizing the spatial, temporal and spatio-temporal features of eye-tracking data [35]. Many of these representations are based on the notion of *scan-path*, i.e., the path of oculomotor events in space recorded during a certain period of time [19].

However, with the increasing complexity of the stimulus and the high number of fixations recorded throughout several eye-tracking trials, typical visualizations become cluttered and hard to understand. To overcome this issue, the stimulus can be segmented into *Area(s) of Interest* (AOI), which define the regions of the stimulus that are susceptible to provide meaningful insights to researchers. These regions are usually used to investigate the focus on specific parts of the stimulus [19]. Eye-tracking data can be grouped by areas of interest and represented as *dwells*. A dwell refers to one visit in an AOI from entry to exits points [19]. A common metric associated to AOIs is the dwell time, i.e., the time gazed at a certain AOI, from entry to exit [36], including all fixations and saccades landing within the AOI. The total dwell time is the sum of all dwell times to a specific AOI over a trial, including revisits on the same AOI. Representing eye-tracking data as dwells allows describing the visual behavior in a more concise way, based on the visited AOIs.

The eye-tracking literature proposes different techniques to visualize the way people engage with visual stimuli [23]. These visualizations provide insights about common strategies and patterns followed by the humans when interacting with the stimulus. In the context of this study, we deploy two visualizations: attention maps and scarf-plots. Attention maps [37] are aggregation of fixations over time and/or over participants [23] and are often used as an alternative for transitions matrices to capture transitions between AOIs. Unlike transition matrices that emphasize only the spatial feature of AOIs [38], attention maps consider also the ordering dependencies between AOIs. To obtain these maps, fixations are grouped by dwells, then direct relationships between dwells are identified and used to construct a dependency graph illustrating reading patterns. Attentions maps provide important insights about the dwell time and the transitions between AOIs. However, the generated dependency graphs are unable to describe the evolution of the reading patterns over time. Instead, scarf-plots [19] provide a timeline representation allowing to observe the changes in the visited AOIs over time, which in turn could provide pertinent insights about the users' visual search behavior.

The literature discerns two types of search behaviors, namely *goal-directed search* (also referred to as "task-oriented processing" in [39]) and *exploratory search* [20, 24]. Goal-directed search occurs when users follow a certain search routine to gather general or specific information to solve a task. Oppositely, exploratory search occurs when users lack the experience to search efficiently and rather screen the environment without having a clear search plan in mind [20]. According to a recent study about consumers' attention processes [25], the total number of different products that are fixated by a user and the average time that the user spends fixating detailed product information can be used to distinguish between goal-directed and exploratory search. The comprehension of process models comprises an active search for relevant information [40], which would be interesting to scrutinize in the light of the search behaviors reported in the literature.

3 Related Work

This section presents the works related to this paper. Section 3.1 introduces existing research on hybrid process artifacts, Section 3.2 describes the eye-tracking studies in the context of process modeling, and Section 3.3 summarizes behavioral analyses conducted in the field process modeling. Finally, Section 3.4 highlights the novelty of our work compared to previous research.

3.1 Related Studies on Hybrid Process Artifacts

The literature comprises a wide range of studies evaluating the understandability (e.g., [41]), maintainability (e.g., [42]) and modeling (e.g., [43]) of business processes. However, with regards to hybrid process artifacts, only a handful of empirical studies exist (for a systematic literature review see [11]). In [4, 5], Zugal et al. introduce the Test Driven Modeling (TDM) approach. This hybrid approach combines a declarative process model (modeled in Declare [28]) with test cases to support users when interpreting declarative models. The hybrid process artifact proposed by the authors was evaluated in two empirical studies [44]. The results show that the use of test cases as a hybrid process artifact provides an additional communication channel and supports the maintainability of declarative process models by lowering the participants' cognitive load and increasing the perceived model quality.

De Smedt et al. [6, 7] propose a hybrid process artifact combining a declarative process model with textual annotations describing the hidden dependencies in the model. In this work, the authors address the issue of hidden dependencies and propose a methodology allowing to infer them and make them explicit for the users. Following a quantitative approach, the authors evaluate the understandability of the proposed hybrid process artifact. The results support the hypothesis that the proposed hybrid representation contributes to better comprehension accuracy and reduced response time and cognitive load when compared to conventional (i,.e., non-hybrid) declarative representations.

Along the same lines, Lopez et al. [8] propose the process highlighter as a means to interlink the constraints in the process specifications with the constructs of the process model. The tool is embedded in a process artifact to support modelers during process modeling. The hybrid process artifact is evaluated by Andaloussi et al. [9]. Following a qualitative approach, the authors investigate the way users engage in a modeling task with the support of the process highlighter. The results suggest evident support during process modeling and highlight several benefits resulting from the explicit mapping between textual process specifications and modeling constructs, which in turn could improve the quality of designed process models.

3.2 Eye-Tracking in Process Modeling Research

Eye-tracking is used in different fields to study humans' behavior and cognition [19, 45]. In the field of process modeling, eye-tracking is used to study the understandability of process models and to predict users' performance in solving comprehension tasks. In particular, Petrusel and Mendling [46] investigate the impact of focusing on relevant regions on the comprehension of BPMN [47] process models. The authors formalize the notion of relevant regions, derive new metrics based on the notions of precision, recall, and their harmonic mean, and propose a statistical model allowing to predict the answer accuracy of participants based on such relevant regions.

Similarly, Bera et al. [48] study how attention on relevant parts of the process models influences users' performance, and scrutinize the visual association of modeling constructs during a comprehension task. Following a set of quantitative and qualitative studies supported by eye-tracking and concurrent think-aloud, these two research aspects are investigated considering different notations: BPMN [47], EPC [49] (Event-driven Process Chains) language, and EPC-H (i.e., a variant of EPC where roles are highlighted in different colors). The quantitative analysis shows that both pools and lanes in BPMN [47] and the highlighted roles in EPC-H help participants to focus on the relevant parts of the model, whereas EPC without role-highlighting fails to support that. As for the qualitative analysis, the results allow the authors to link the visual associations between different parts of the model (seen with the support of eye-tracking) and the underlying cognitive integration processes (articulated through concurrent think-aloud).

Empirical evaluations of process models using eye-tracking also cover models integrating business rules. Driven by the principle of separation of concerns and the need to integrate business rules in process models in an understandable manner, Wang et al. [50] investigate the effect of linked rules (i.e., a specific class of business rules) [51] on the understandability of process models following a quantitative approach. The outcome of the statistical analysis demonstrates that the integration of linked rules with process models is associated with decreased response time and reduced cognitive load, while a positive impact on the comprehension of the model remains partially supported.

The experience with the use of eye-tracking in process modeling is investigated by Zimoch et al. [52]. In this work, the authors report the lessons learned from a study investigating the comprehension of process models in different imperative notations. The findings are summarized into a set of recommendations meant to improve the design of eye-tracking experiments. Overall, these recommendations address the familiarity of participants with the process scenario and their prior expertise with the modeling language, which could potentially bias the validity of the results. In addition, the authors encourage combining eye-tracking measures with other cognitive biosensor-based measures and suggest that the results should be interpreted in light of the existing theories in cognitive psychology.

3.3 Behavior Analysis of Humans Engaging with Process Models

While several empirical studies investigate the comprehension of process models, only a few of them analyze the behavior of participants when interacting with these models. Haisjackl et al. [53] investigate the reading of declarative process models (in Declare [15]) following a qualitative approach supported by concurrent think-aloud. The analysis of the verbal protocols reveals important insights about the way the participants engage with the provided process models. In particular, the results show that the participants read Declare models in a sequential way, usually refer to the left corner activity to identify the entry-point of the model, follow a top-down strategy to read hierarchical processes, and ignore hidden dependencies when describing the model.

In another context, Haisjackl et al. [54] investigate the way imperative process models (expressed as BPMN) are inspected for quality checks following a qualitative approach supported by concurrent think-aloud. The analysis of the verbal data allows the identification of different strategies used for inspecting process models and sheds light on the order in which quality checks are generally performed.

Regarding the modeling of business processes, the PhD work of Pinggera [55] provides important insights about the behavioral patterns of modelers during process modeling. Following a qualitative approach supported by concurrent think-aloud and user interactions, the author identifies the order in which different modeling phases occur (e.g., problem understanding, method finding, modeling, reconciliation, validation) and proposes a catalog of modeling patterns. In a follow-up study, the author uses a quantitative approach to derive a statistical model supporting the identification of different modeling styles [56].

3.4 Novelty Compared to Previous Research

This paper differs from earlier studies in several aspects. As opposed to [6, 7, 44] our work takes a different approach to look at the comprehension of hybrid process artifacts by analyzing the participants' behavior when being confronted with hybrid process artifacts, which in turn allows perceiving the usability of hybrid process artifacts from a different perspective. With regards to [46], our interest does not lie in the definition or discovery of quantitative metrics for evaluating comprehension performance, but we rather focus on identifying general strategies that are descriptive of users' behavior in the context of process model comprehension. Moreover, while [46] focuses on quantitative findings, we base our analysis solely on qualitative data, complementing the findings derived from eye-tracking analysis with think-aloud data. Compared to [53,54], we consider eye-tracking data collection and analysis, thus adding a novel dimension to describe users' behavior. Last but not least, to the best of our knowledge, this is the first paper providing insights on how people read and interact with DCR Graphs alone and with DCR-HR.

4 Research Method

This section describes the steps followed to design the exploratory study and to analyze the collected data. In Section 4.1, we provide an overview about the design and the analysis of our study. Afterwards, in Section 4.2 we highlight the key aspects of the study design and execution, while in Section 4.3 we focus on the different approaches used for data analysis.

4.1 Overview

Our research aims at exploring how domain experts and IT specialists engage with hybrid process representations, in particular DCR-HR. To this end, we designed an exploratory study supported by eye-tracking and think-aloud and recruited municipal employees (serving as proxies for domain experts) and academics (serving as proxies for IT specialists). After the data collection, we conducted a multi-granular qualitative analysis to gain insights about the users' behavior when interacting with the different DCR-HR artifacts (i.e., DCR graph, law text, simulation).

Table 1 presents the goal of our study following the Goal, Question, Metric (GQM) template [57]. To address this goal we formulated three different research questions and used several eye-tracking measures and qualitative codes as indicators to understand the behavior of participants. Overall, we looked into the way participants with different background engage with DCR-HR when solving different tasks, assessed the benefits and challenges associated with each of the DCR-HR artifacts and explored the different search strategies used by the participants when interacting with DCR-HR. In the following (Sections 4.2 and 4.3), we describe the design and execution of our study and explain the different analysis approaches pursued to infer and use the measures and indicators presented in Table 1.

4.2 Exploratory Study Design and Execution

To gain insights into the use of hybrid process artifacts (specifically DCR-HR) during comprehension tasks, we design an exploratory study that collects and analyzes eye-tracking and retrospective think-aloud data.

Exploratory study design. As previously introduced, DCR-HR encompasses three different process artifacts, namely a DCR Graph [17], a textual process specification based on excerpts of the law and an interactive simulation (cf. Section 2.2). Such artifacts are meant to support stakeholders with different backgrounds when performing different types of tasks. Indeed, while domain experts are typically knowledgeable with the law, IT specialists are rather familiar with process models. Since little is known about how people engage with hybrid process artifacts, analyzing user behavior would provide important insights about the way the different DCR-HR artifacts are conjoined and used during a comprehension task. This, in turn, could help developers to improve the current tool based on personal user preferences. Moreover, literature [2,3,44] suggests that hybrid process artifacts have the advantage to provide different perspectives on the process which might be purposeful to a different extent depending on the specific task to be addressed. Thus, it is important to scrutinize whether they are used in the same way when performing different tasks. Such input is expected to further improve tool-support based on the characteristics of the running task and to provide ad-hoc recommendations to support each user based on his or her background. Based on these considerations we formulate the first research question as follows: **RQ1 - How do users engage with the different DCR-HR artifacts?**

In addition, to gain a deeper understanding of why users interact with DCR-HR in a certain way, we examine the perceived benefits and challenges associated with their use, keeping in mind that each artifact conveys information that can be used independently or combined with the one carried by other artifacts to pursue a certain goal. By collecting the subjective insights of the participants, we are also able to support the interpretation of the results related to RQ1. Hence, the second research question can be stated as follows: **RQ2 - What are the benefits and challenges associated with each one of the artifacts of DCR-HR?**

Finally, we are interested in exploring how the use of different artifacts evolves over to time. The temporal sequencing of eye movements allows us to identify common strategies followed to approach comprehen-

Goal	**Purpose** **Issue** **Object** **Viewpoint**	Explore the way users engage with DCR-HR from the domain experts and IT specialists viewpoints
Question **Metrics**	RQ1	How do users engage with the different DCR-HR artifacts? Mean fixation duration and mean transition frequency
Question **Indicators**	RQ2	What are the benefits and challenges associated with each one of the artifacts of DCR-HR? Qualitative codes emerging from the verbal utterances of participants
Question **Metrics**	RQ3	What strategies are followed when engaging with the different DCR-HR artifacts? Durations of fixations on relevant AOIs and total number of fixated AOIs

Table 1 GQM Model describing our goal, research questions and used metrics and indicators.

sion tasks. Accordingly, we formulate the third research question as follows: **RQ3 - What strategies are followed when engaging with the different DCR-HR artifacts?**

To explore the previously introduced research questions, we begun with collecting the gaze data of participants interacting with DCR-HR during process model comprehension tasks and then collected their subjective insights during a retrospective think-aloud session.

The context of this study is the digitalization of the law and, in particular, we focus on a process derived from paragraph §45 of the Danish "Consolidation Act on Social Services" (cf. Section 2.2).

Participants were 5 municipal employees from Syddjurs Municipality in Denmark and 10 academics studying or working at the Technical University of Denmark or at the IT University of Copenhagen. The municipal employees (who serve as proxies for domain experts) had proficiency in reading legal documents, but had (with one exception) no prior knowledge in process modeling or IT development. Academics (who reflect the common profiles of IT specialists) had background in process modeling and IT development, but lacked experience in reading and interpreting legal texts. Although, all academics have worked with process models in the past, not all of them were familiar with the DCR notation.

The data collection phase was organized into six comprehension tasks presented in the form of questions that participants had to answer using the different artifacts of DCR-HR. Following the input of experts in the field of process modeling, the comprehension tasks were designed to be easily grouped into three categories reflecting typical situations that a user may face when dealing with a process model.

The first group of *constraint tasks* comprises questions about the relationships between pairs of activities represented in the process model. These questions reflect a typical process comprehension task as a user is expected to have a clear understanding of which are the activities and the constraints that are relevant to the question to provide an answer. The second group of *decision tasks* comprises questions prompting the user to decide among multiple options. In this regard, a user is expected to identify the contextual information required to guide his or her decision-making process to achieve the desired outcome. Such contextual information is often not included in the process model and, thus, a user is expected to make a decision by relying on the process specification, i.e., the law text (cf. Section 2.2). Finally, *scenario tasks* concern the execution of partial process instances and comprise questions asking a user to determine whether or not a certain behavior is admissible based on a given case history. Scenario tasks are typical of process model testing and validation. Indeed, to validate a process based on positive and negative test scenarios, it is necessary to keep all execution traces in mind in order to check whether a certain behavior is admissible or not.

The comprehension tasks were designed and displayed on the DCR Graphs Portal [18] either in English or in Danish depending on the individual preference of the participants. Visually, the content of the web-based user interface was organized into the four main areas outlined in Fig. 1 (cf. Section 2.2).

The complete material designed for this study can be found online[4].

Executing the exploratory study. The execution of the exploratory study was organized into a PREPARATION PHASE and a DATA COLLECTION PHASE, which were carried out individually for each participant and whose details are captured by the BPMN [47] process of Fig. 2.

At the beginning of the study, each participant went through a PREPARATION PHASE consisting of the following four steps (cf. the first four tasks of the process of Fig. 2) a physical assessment questionnaire, a background data questionnaire, an introduction to DCR-HR

---

[4] `http://andaloussi.org/sosym2019/design/`

and to the DCR Graphs Portal, and a final calibration step.

The Physical assessment questionnaire was carried out to check the physical ability of a user to participate in a video-based eye-tracking experiment. Indeed, people wearing glasses or contact lenses, or having reduced vision may compromise the acquisition of high-quality eye-tracking data [36]. Overall, seven of the participants claimed to wear glasses or contact lenses and only one person (not among those wearing glasses) reported having some troubles seeing things located far away. However, since no-one reported any major vision issue, we decided not exclude anyone from the study at this point [36]. The physical assessment questionnaire and data are available in our online appendix.[5]

Afterwards, we administered a Background and expertise questionnaire asking each participant some questions about (i) his or her ability to read and create DCR Graphs and to use the interactive simulation, (ii) his or her familiarity with the process used in the comprehension tasks. With regards to (i) the collected data show that academics were indeed more familiar with reading, creating and simulating DCR Graphs than municipal employees (averages on a scale of 5: 3.9 and 1.8 familiarity with reading DCR Graphs respectively, 2.3 and 1.2 familiarity with creating DCR Graphs respectively, 3.6 and 1.2 familiarity with simulating DCR Graphs respectively). Concerning the familiarity with the considered process (ii) only one participant claimed to be already familiar with the considered process. The background and expertise questionnaire and data are available in our online appendix[6].

Then, we provided a 30-minute introduction to DCR-HR, going in detail through the semantics of the different DCR Graph relations [17], and presented the layout and the main features of the DCR Graphs Portal (cf. task Introduction to DCR-HR and portal in Fig. 2).

As a last preparation step, we conducted a hardware calibration procedure to ensure good data quality (cf. task Calibration and testing in Fig. 2). In detail, we used a 9-points calibration and tested its accuracy with all participants prior to starting with data collection.

The first comprehension task (i.e., Familiarization task in Fig. 2) was designed to allow participants get acquainted with the DCR Graphs Portal. Then, the remaining six comprehension tasks were displayed sequentially and the user could proceed in a self-paced way (participants took an average of 15 minutes to execute all the comprehension tasks). During the execution of the COMPREHENSION TASKS, we recorded eye gaze data using the Tobii Pro X3–120 eye-tracker[7], which was placed in front of the monitor that presented the comprehension tasks.

At the end, we conducted a Retrospective think-aloud session [19] to collect insights about the use of DCR-HR. During this phase, we recorded all the subjective insights provided verbally by the participants about their experience with DCR-HR. The session was organized as a guided interview structured as a set of open-ended questions meant to help the participants to articulate the adopted problem-solving strategies and identify the encountered challenges. The following two questions provide a good summary of what was asked to the participant during this last step of the data collection phase: (i) How and for what purpose did you use the following artifacts: model, law text, simulator? and (ii) Thinking about the overall experiment, was there anything challenging? The goal of the think-aloud was to encourage participants to elaborate on specific aspects that were deemed important for the exploratory study. Particularly, with (i) we tried to investigate the perceived benefits associated with each DCR-HR artifacts and to understand how the artifacts have been used and combined to perform different tasks. Similarly, with (ii) we invited participants to reflect on the challenges which could potentially hinder the use of the different DCR-HR artifacts. Both questions gave rise to important insights, which were coded and organized following the principles of grounded theory [58] to serve as a reference for the analysis introduced in the following section.

4.3 Analysis Approach

To answer research questions RQ1–RQ3, we followed different approaches to analyze the eye-tracking and think-aloud data collected during the exploratory study.

Eye-tracking data analysis. Eye-tracking is a widely used method to reveal patterns of visual behavior of humans [20] and provides insights into several information processing tasks [59].

Fig. 3 depicts the general ideas behind the eye-tracking data analysis followed in this paper.

Usually, the raw gaze data recorded by eye-tracking devices are aggregated into fixations and saccades, i.e., the two major categories of oculomotor events used in the analysis of eye movements [19]. In this study, we

---

[5] See `http://andaloussi.org/sosym2019/design/Physical_assessment_questionnaire.xlsx`

[6] See `http://andaloussi.org/sosym2019/design/Background_expertise_questionnaire.xlsx`

[7] See `https://www.tobiipro.com/product-listing/tobii-pro-x3-120/`

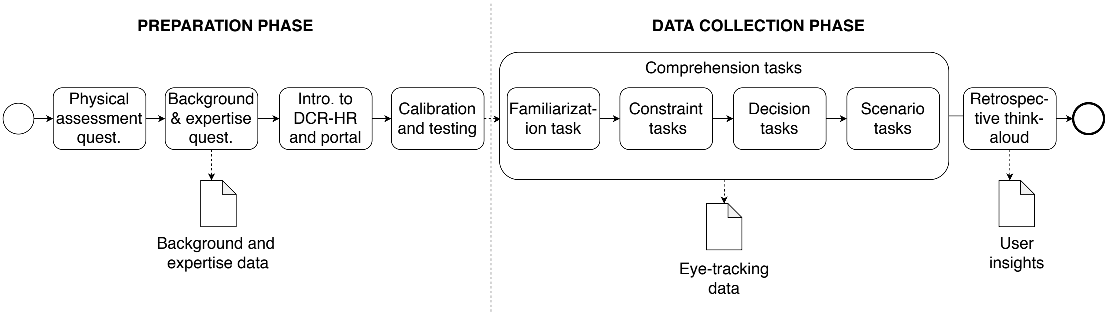


Fig. 2 BPMN process summarizing the PREPARATION and DATA COLLECTION phases of the conducted exploratory study. Collected data are captured by BPMN data objects.

derived fixation data from gaze points [19] using the I-VT Algorithm [60] implemented in the Tobii Pro Studio 3.4.8 software (cf. step ① in Fig. 3).

Then, prior to diving into the analysis, we looked into factors potentially affecting the quality of the collected data [36]. To this end, we replayed the eye gaze recordings to qualitatively assess the accuracy of the collected eye-tracking data for each individual. We also relied on the data quality measures provided by the eye-tracking software, e.g., the proportion of time spent on fixations compared to the total time spent executing the task. As a result of this data quality assessment step, we excluded two participants from the analysis. One participant, whom we labeled P04 was removed because of missing data: indeed, the average fixation time per comprehension task amounts to 02.53 seconds, which is much shorter than the 69.06 seconds spent by the other participants on average. This participant was the one claiming to have prior knowledge with the process, which may explain the reduced amount of fixations. Another participant, whom we labeled P10, reported having issues with peripheral view forcing him to rotate his head to read. Indeed, by replaying the gaze recordings, it is clear that the data collected for P10 are not accurate. Although the calibration was successful for both P04 and P10, it is likely that the mapping between their gazes and the corresponding coordinates on the stimulus failed at some point during the data collection.

Then, since our research questions consider different DCR-HR artifacts, which are located in a specific position of the screen, we labeled fixation data based on the targeted Area Of Interest (AOI). In this way, we were able to analyze the amount of time that each participant spent fixating a particular artifact of DCR-HR.

When defining areas of interest, we considered two levels of granularity. First, to answer **RQ1** at a coarse-grained level, we considered 3 areas of interest, each one referring to a distinct DCR-HR artifact, i.e., the DCR Graph, the law text and the simulation (cf. Fig. 1 and step (2a) in Fig. 3). Then, to enable a more fine-grained data analysis and look into users' behavior (cf. **RQ3**), we divided the AOI framing the DCR Graph into 22 smaller AOIs, each one referring to a distinct model element (i.e., activity or relation of the graph). We also defined a novel AOI including the question title, obtaining 25 AOIs overall as shown in Fig. 4 (cf. step (2b) in Fig. 3).

After having defined the AOIs, we exported from the eye-tracking software time-stamped data sets of fixations including an identifier for each participant, the considered comprehension question, the duration of the fixation and the AOI hit, which defines whether a fixation falls within a certain AOI or not [23] (cf. steps (3a) and (3b) in Fig. 3).

To analyze the obtained fixation data and explore research questions **RQ1** and **RQ3**, we followed two distinct approaches, based on the granularity used in the definition of the AOIs and on the analysis goal.

For the first analysis, we used the three coarser AOIs, exploited process mining [22] and AOI-based attention maps [19] to explore the relationships between different AOIs. This approach was followed to investigate **RQ1**.

As a first step, we transformed fixation data into an XES event log [61] by merging contiguous fixations referring to the same AOI (cf. step (4a) in Fig. 3). Then, after identifying the directly-follow relationships in the log [62], we generated a descriptive process model (referred to as "attention map" in the context of this work) illustrating how the attention of participants was distributed among the three artifacts of DCR-HR (cf. step (5a) in Fig. 3).

An example of such attention maps is depicted in Fig. 5. In the proposed graphical representation of at-

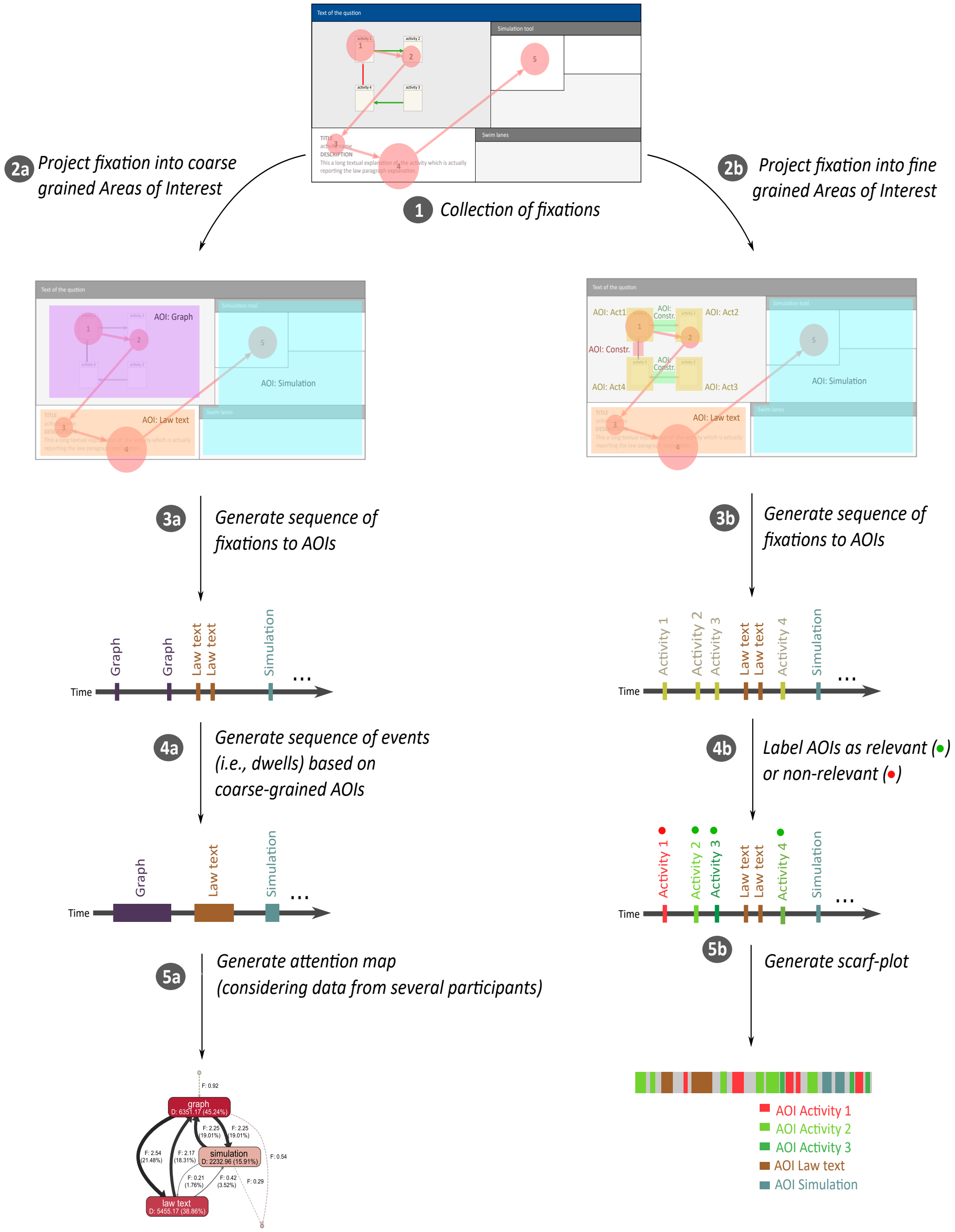
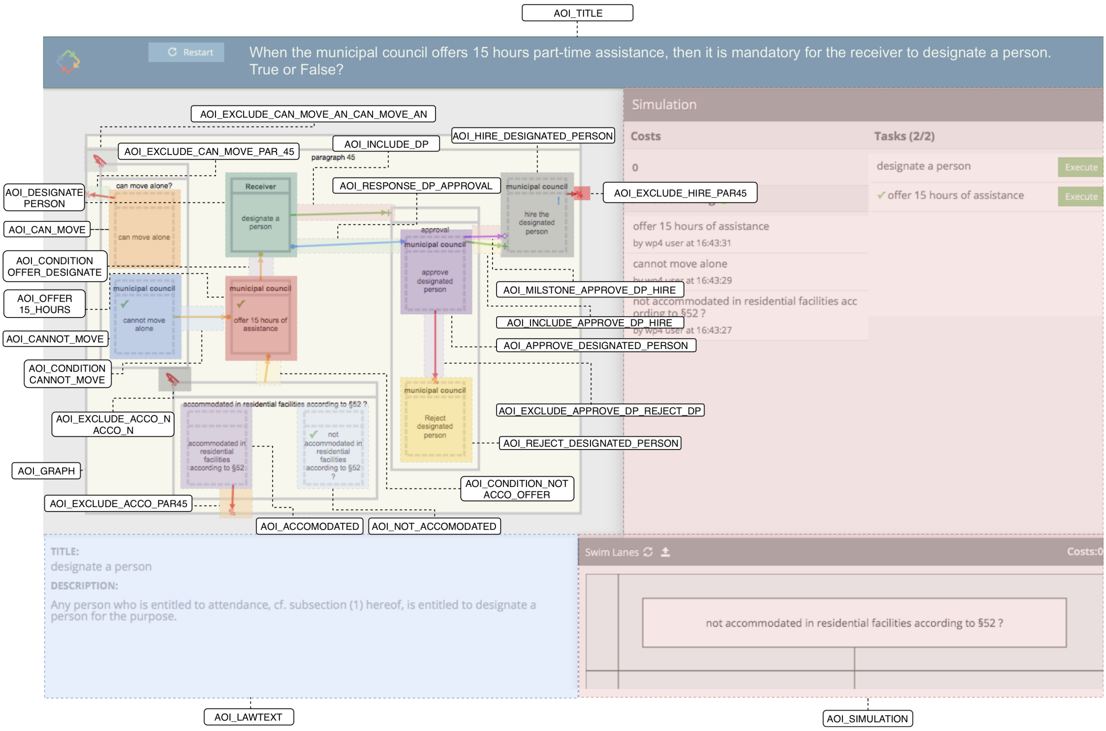


Fig. 3 Overview of the two eye-tracking data analysis approaches followed in this paper.


When the municipal council offers 15 hours part-time assistance, then it is mandatory for the receiver to designate a person. True or False?

Simulation

Costs	Tasks (2/2)
0	designate a person Execute
	✓ offer 15 hours of assistance Execute

offer 15 hours of assistance
by wp4 user at 16:43:31

cannot move alone
by wp4 user at 16:43:29

not accommodated in residential facilities according to §52 ?
by wp4 user at 16:43:27

TITLE:
designate a person

DESCRIPTION:
Any person who is entitled to attendance, cf. subsection (1) hereof, is entitled to designate a person for the purpose.

Swim Lanes Costs:0

not accommodated in residential facilities according to §52 ?

Fig. 4 Overview of the 25 areas of interest used for fine-grained eye-tracking data analysis. A higher resolution of this figure is available at `http://andaloussi.org/sosym2019/figures/DCR-HR-FineGrainedAOIs.pdf`

tention maps, the considered AOIs, corresponding to the three artifacts of DCR-HR, are represented as activities, whereas the transitions between them are represented as edges. Activities are labeled with the name of the artifact they represent, the mean dwell time (D), and the proportion of the overall time spent focusing on that artifact, which is represented as a percentage value and visualized through the intensity of the background color used for the activity. Relationships are labeled with the mean transition frequency (F) and with a percentage value quantifying the number of times that transition occurs compared to the total number of transitions. Graphically, this is rendered by the thickness of the arrows sketching the relationships. Finally, we represent the starting and ending points of the task and draw a dashed arrow connecting them to the first, respectively, last fixated artifacts.

In order to analyze the reading patterns of participants, we extracted the total fixation duration on each AOI (i.e., the dwell time [19]) and the frequency of transition between each pair of AOIs. Afterward, the mean fixation duration (D) and mean transition frequency (F) were derived by dividing each measure by the number of traces (i.e., sequences of dwells) used to discover the attention map. These two measures were projected respectively on activities and edges in the attention map to compare the reading patterns in different attention maps.

The second, more fine-grained analysis, followed to investigate **RQ3**, explores all the 25 defined AOIs and also considers the temporal dimension of fixation data, grounding on timeline AOI-based visualization approaches [23].

Specifically, after exporting the sequence of fixations (cf. step ⓷b in Fig. 3) we examined how the obtained sequences of visits unfolded along a timeline skipping the aggregation into dwells done in the previous analysis (cf. step (4a) in Fig. 3). Indeed, this time we are interested in keeping a fine-grained resolution of fixations to know when and for how long a user fixated a certain AOI. This approach is beneficial to know when AOIs are visited during task execution and to detect consecutive revisits on the same AOI, keeping track of the saccades occurring between them.

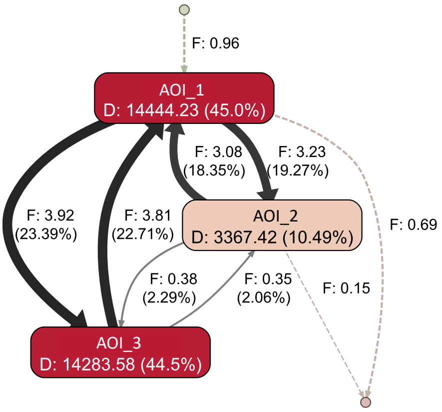


Fig. 5 Example of attention map showing three AOIs and the transitions among them. D represents the mean fixation duration while F is the mean transition frequency between two AOIs.

During this phase, we also labeled the AOIs of the DCR Graph based on whether they were deemed "relevant" for answering a particular question (cf. step (4b) in Fig. 3). Indeed, some comprehension tasks require the user to look for information that is explicitly represented in the DCR Graph and to combine it with contextual (law text) or execution (simulation) information to provide an answer. The AOIs related to the law text and the simulation have not been explicitly labeled as "relevant" or "non-relevant", but their relevance has been considered based on the kind of executed task.

Then, to visualize the visits on different AOIs over time we relied on scarf-plots [19, 59] (cf. step (5b) in Fig. 3), which show a timeline for each participant (i.e., a "scarf") divided into colored time spans. Each different color is used to encode a specific AOI, and the width of the time span is proportional to the duration of the fixation on that AOI. This visualization technique is particularly efficient for exploring and comparing scan-paths and sequences of dwells, as it shows which AOIs have been fixated often and when during task execution [23]. When creating the scarf-plots, we took inspiration from the qualitative alphabet palette of Polychrome, which includes 26 colors that are well separated in the CIE L*u*v* color space [63] and used all the colors in the palette apart from those in the range of greens. Indeed, a range of greens was created on purpose to be assigned to all the relevant AOIs in the graph to ease their identification during the analysis.

To create and visualize the scarf-plots used for the analysis, we adapted the piano-roll view of the Rythm-Eye tool [64]. Fig. 6 shows an example of scarf-plot depicting the sequences of fixations for three participants A, B and C. Each timeline captures the visual behavior of each participant over time. Colored time spans correspond to the visited AOIs: relevant AOIs are assigned a color belonging to the range of greens and are labeled in italic. It is worth noticing that the scarf-plot of Fig. 6 is normalized, that is, all the scarves have the same width regardless of the total time spent by each participant to execute the task. In this study, we relied on normalized scarf-plots as they facilitate the comparison of the visual behavior of different participants.

The analysis of scarf-plots was aimed to explore the users' visual behavior and to identify common strategies followed to engage with DCR-HR. To this end, we followed a qualitative coding approach. First, the scarf representing the behavior of each participant was inspected individually by appending qualitative memos [65] describing the observed behavior. During this process we relied on the distinction between goal-directed search and exploratory search introduced in the literature (cf. Section 2.3) to classify the participants' behaviors. We used two eye tracking measures (i) the time spent fixating AOIs that are "relevant" for executing the task (i.e., relevant areas of the DCR Graph and, possibly, the law text and the simulation based on the kind of executed task) and (ii) the total number of AOIs fixated during task execution. We assigned the label *goal-directed search* to the participants who focused on task-relevant information and visited only a small number of AOIs, while, we assigned the label *exploratory search* to the participants who switched their attention between a large number of AOIs, without showing particular interest for a specific AOI. We supported our labeling by triangulating the insights reported in the memos with the verbal data extracted from the think-aloud sessions, the eye gaze recordings, and some of the descriptive statistics inferred from the eye-tracking data. During this phase, co-authors dis-

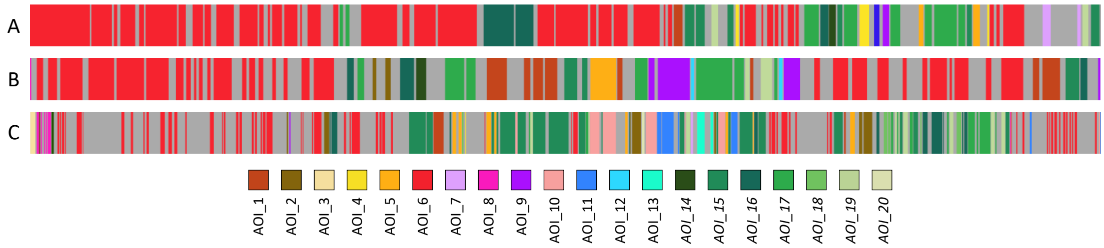


Fig. 6 Example of scarf-plot showing the fixations of three participants over time. Grey areas indicate that no AOI has been fixated at that particular moment.

cussed borderline cases to reach a consensus on how to classify participants exhibiting a potentially "mixed" behavior. Indeed, by going through gaze recordings, we noticed that even within the same task, some participants used more than one strategy when reading the law text or the DCR Graph.

Think-aloud data analysis. To answer **RQ2** we relied on think-aloud data. Prior to beginning with the analysis, we transcribed the verbal data collected during the retrospective think-aloud. Afterwards, we extracted the subjective insights from the transcripts with the support of Atlas.ti[8], a qualitative data analysis software tailored to deal with large bodies of textual, graphical, audio and video data [66]. During this process, we applied coding concepts from grounded theory [58], a methodology for developing theory by analyzing a text corpus, iteratively identifying recurring aspects and grouping them into categories. In particular, we used initial-coding to fragment the textual data and identify the emerging topics raised by the participants. Afterwards, we applied focus coding to identify recurring codes representing significant aspects into categories. Finally, we followed the principles of axial coding to find relationships among the inferred categories. The emerged codes serve as indicators of the use of the different DCR-HR artifacts and provide deeper insights about the patterns found in the eye-tracking data.

The complete analysis material can be found online[9]

5 Findings

This section reports the main findings of the exploratory study, presented according to the research question they concern. Section 5.1 presents the results related to the coarse-grained eye-tracking analysis. Section 5.2 reports the benefits and challenges associated with each of the DCR-HR artifacts. Finally, Section 5.3 describes the results of the fine-grained eye-tracking analysis.

5.1 How Do Users Engage with the Different DCR-HR Artifacts? (RQ1)

This section describes the results of analysis conducted to explore how people use DCR-HR. In particular, we explore whether the use of the different DCR-HR artifacts is influenced by (i) the participants' background or (ii) the kind of performed tasks.

(i) First of all, we begin with differentiating between municipal employees and academics based on their different backgrounds. Indeed, municipal employees are accustomed to reading legal texts but lack knowledge in process modeling, whereas academics have experience in process modeling but lack proficiency in legal reading. Based on their different backgrounds, we expect municipal employees and academics to prefer different artifacts (in line with their background) while performing the comprehension tasks.

To explore this assumption, we used both attention maps generated from the eye-tracking data and aggregated over all participants and the insights derived from the think-aloud data. Fig. 7a and Fig. 7b show the attention maps comparing the reading patterns of municipal employees and academics when answering the given tasks.

By taking a closer look at them, we can see that both groups started by observing the DCR Graph, which is reasonable since the DCR Graph is placed in the center of the screen and occupies a large portion of it (cf. Fig. 1). Furthermore, we can observe that academics spent substantially more time looking at the different artifacts compared to municipal employees (cf. mean fixation duration D in Fig. 7a and Fig. 7b). This observation is also supported by the subjective insights retrieved from the think-aloud transcripts. Indeed, municipal employees reported to have relied on common

---

[8] Atlas.ti, a qualitative data analysis tool. See `https://atlasti.com`

[9] see `http://andaloussi.org/sosym2019/analysis/`

sense or on their working experience when answering to some tasks (e.g., *"If the recipient is unsatisfied, then, of course, you can change the decision* [while the DCR Graph shows clearly that such a decision cannot be reversed]*"*[10]). Instead, academics claimed to have never relied on common sense when performing the tasks.

The attention map depicted in Fig. 7a reveals that most municipal employees split their attention mainly between the DCR Graph and the law text. From the think-aloud, we found evidence that the majority of municipal employees did not use the simulator, but relied only on the graph and the law text (e.g., *"I have either read through the law text or the model but I have not used the simulation."* [10]). However, some municipal employees used all the three artifacts. In particular, a municipal employee affirmed to have used the law text but having relied on the simulation to validate his answers, while another municipal employee mentioned using the simulation twice during the whole experiment. These insights line up also with the proportions of transitions between the artifacts. Indeed, Fig. 7a shows that the number of transitions between the graph and the simulation and between the graph and the law text is similar, suggesting that municipal employees have generally interacted with all the different artifacts.

When looking at the attention map of Fig. 7b, outlining the use of artifacts carried out by academics, we can observe that artifacts have been used differently. Academics focused mostly on the DCR Graph and split the rest of their attention between the law text and the simulation. Accordingly, the proportion of transitions between the different artifacts shows that academics did almost twice more transitions between the graph and the simulation than between the graph and the law text. Indeed, academics not only spent a limited time on the law text compared to municipal employees but have switched less often between the DCR Graph and the law text. By looking into think-aloud data we had confirmation that many of the academics struggled to understand the legal terms and the linguistic patterns used in the law text (e.g., *"I think understanding this law jargon was kind of difficult"*, *"I tried to read the law text to understand the law but it actually didn't help at all because the language that is used is pretty formal"*) and, therefore, may not have found it useful.

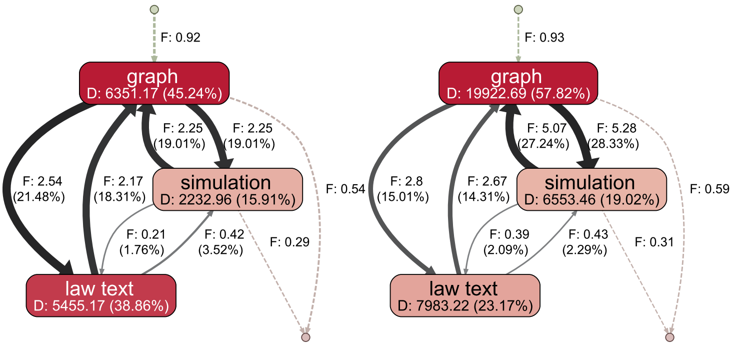
Overall, the attention maps shown in Fig. 7 suggest that users with different backgrounds would use different artifacts to understand the process. Hereby, the hybrid nature of DCR-HR can provide a unified representation that can make process models accessible for users with different backgrounds.

(ii) Besides looking into how the users' background affects the use of different artifacts, we investigated whether the choice of which artifacts are used changes when dealing with different tasks. To explore whether constraint, decision, and scenario tasks (cf. Section 4.2) are executed with the help of different artifacts, we rely again on attention maps.

Fig. 8 depicts the attention maps summarizing how all participants used the artifacts of DCR-HR when executing constraint, decision and scenario tasks respectively. These visualizations reveal a different use of artifacts for each type of task. Fig. 8a shows that in constraint tasks (i.e., questions asking to focus on the relationships between pairs of activities of the DCR Graph), the participants focused mostly on the DCR Graph, and split the rest of their attention between the simulation and the law text. This use of the graph can be explained by the nature of constraint tasks and it is confirmed by some subjective insights obtained from think-aloud data. However, some other participants seemed to be challenged with the semantics of DCR relations and were often resorting to the simulation to clarify the implications that the different relations have on the model behavior (e.g., *"The simulator, I used it when I was in doubt because, the different arrows I wasn't always sure what they did, so then I rendered simulator . . . then you could actually know for sure if you could do this after this or not"*). These subjective insights find confirmation in the high number of transitions between the DCR Graph and the simulation. Indeed, the participants did twice as many transitions between these two artifacts than between the graph and the law text.

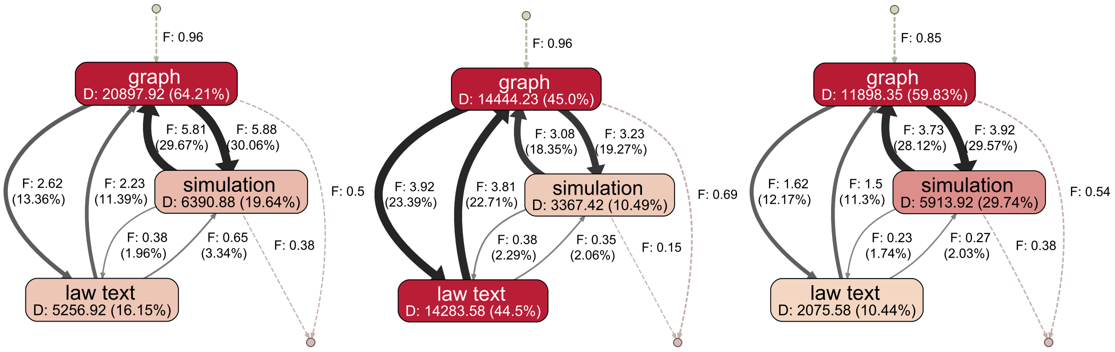
When examining the attention map related to decision questions, shown in Fig. 8b, we can observe that participants split their attention mainly between the DCR Graph and the law text. This evidence suggests that, when asked to choose among multiple options, participants relied on the law text to retrieve contextual information useful to support their decision-making process. This explains also the high number of transitions between the graph and the law text occurring in decision tasks.

Finally, Fig. 8c shows how the different artifacts have been employed in scenario tasks. By looking at the attention map, we can clearly see that the participants spent relatively less time on the law text while switching their attention mainly between the DCR Graph and the simulation. Hereby, one can argue that the participants have mainly combined the DCR Graph and the simulation to answer scenario tasks. This assumption is supported by the think-aloud data where participants affirmed using both the DCR Graph and the simulation when asked to determine the behavior of the process model (e.g., *"When the question is in a scenario then I*



(a) Municipal employees (b) Academics

Fig. 7 Attention maps comparing the attention focus on different artifacts for municipal employees and academics. D is the mean fixation duration and F is the mean transition frequency between two AOIs.



(a) Constraint tasks (b) Decision tasks (c) Scenario tasks

Fig. 8 Attention maps showing the use of different artifacts for all participants executing different types of tasks. D is the mean fixation duration and F refers is the mean transition frequency between two AOIs.

use the simulator, because it's easy to see what happens after").

The attention maps for constraint, decision and scenario tasks share one common trait that is the limited number of transitions between the law text and the simulation. The collected think-aloud data support this evidence, as participants could not find any circumstance where the combination of law text and simulation was beneficial (e.g., *"Simulation and law text doesn't go well together because that if you can actually solve it with the simulator, you don't need the law text"*).

Overall, the insights outlined in Fig. 8 show that the participants combined different artifacts when executing different tasks. On this ground, we argue that the deployment of hybrid process artifacts such as DCR-HR can support users dealing with different tasks in a situation-specific manner.

5.2 What Are the Benefits and Challenges Associated with Each One of the Artifacts of DCR-HR? (RQ2)

In this section, we rely on think-aloud data to gather insights into the perceived benefits and challenges of DCR-HR artifacts and to improve our understanding of the use behaviors discussed in Section 5.1.

The results of the think-aloud data analysis show that the DCR Graphs helped several participants to get a good overview about the business process (e.g., *"The model* [...] *I mainly used it to identify how the overall process works"*). The DCR Graph was also used by participants to identify and navigate through the law text (e.g., *"You can highlight different sections of* [the] *law through* [the] *model"*). Some academics reported that the DCR Graph helped them to understand the interplay between the different process activities (e.g., *"I use the model to see* [the] *interaction between the four dif-*

ARTIFACT	BENEFIT	CHALLENGE
DCR Graph	• Provides a good overview of the process • Helps navigating the law text	• The semantics is hard to understand (**ME**) • The model does not capture enough details
Law Text	• Provides more information than the graph • Supports decision-making (**ME**)	• Hard to read (**AC**)
Simulation	• Clarifies the DCR semantics • Allows testing different executions • Can be used to validate assumptions made on the graph	• Inefficient for evaluating all the possible process executions (**AC**) • Time-consuming (**ME**)

Table 2 Challenges and benefits resulting from the analysis of the think-aloud data. The labels **AC** and **ME** highlight whether a certain benefit/challenge was perceived only by academics (i.e., **AC**) or municipal employees (i.e., **ME**). All the other benefits and challenges are perceived by participants belonging to both groups.

ferent activities"), whereas some of the municipal employees seemed to find DCR relations hard to understand and use. These challenges were inferred from the think-aloud, as several participants reported difficulties in identifying the appropriate DCR relation specifying a certain behavior. Last but not least, some participants found the DCR Graph very abstract and pointed out that the model was not capturing all the details specified in the law text (e.g., *"If you only have the model it's very abstract"* or *"The strange thing is that many things which the law is talking about the model did not talk about"*).

The legal text, in turn, provided participants with details which were missing in the DCR Graph (e.g., *"I guess it provided more details in some cases than the model"*, *"The law text might be able to add some details that can't be in the model "*, *"If I didn't think that the model accurately captured enough for me to answer the question, then I would read the whole text instead"*). The participants also mentioned that the law text was effective to support their decision-making process when the DCR Graph allowed for more than one choice (e.g., *"When I had to use the law text, it was for questions about -should I do this at all- for example should I give personal permission, should I take the accept or should I take the reject button on an activity."*).

Several municipal employees expressed a preference for the legal text as they were already familiar with reading and interpreting law paragraphs (e.g., *"I mostly used the law text because that's what I'm used to look at"* [10]). In turn, many of the academics struggled to understand legal terms (e.g., *"I think understanding this law jargon was kind of difficult"*, *"I tried to read the law text to understand the law but it actually didn't help at all because the language that is used is pretty formal"*). For this reason, some academics did not use the law text to gather information about the business process (e.g., *"It is not so easy to read the law text ... I have totally ignored it"*).

Finally the interactive simulation allowed the participants to check the viability of different process executions (e.g., *"The simulation is helpful to see the possible paths"*, *"You can actually see if you have a viable execution"*). Moreover, some academics affirmed that using the simulator helped to reduce the mental effort required to keep track of the all dependencies existing among different DCR relations (e.g., *"It's a little too much to have all the steps in your mind while you're going ... "*, *"It is easier to see it simulated instead of manually analyze the model"*). These comments fall in line with the previous claims about the role of interactive simulations in improving the understandability of declarative process models [4].

The analysis of the transcripts shows also that the simulation helped participants to validate the assumptions they had made by looking first at the DCR Graph (e.g., *"You can like simulate the process then you like get a clear understanding of how the process works ... if you're in doubt of like relations or anything in the graph then you can use the simulation to like confirm what you actually think about the model "*], *"... checking if it is exactly what I thought the model is doing it's actually doing it"*). Yet, other participants pointed out a few drawbacks associated with the use of the simulation. In particular, some academics considered it inefficient having to restart the simulation every time an undesired state was reached (e.g., *"Actually this was not very convenient because you click the all way through and if you miss a click, which I actually did, you need to do it again"*). Others abstained from using the simulation because they were able to mentally simulate the execution of the process (e.g., *"Primarily, I didn't use the simulator at all because I pretty much simulated in my head"*). Municipal employees, who are not used to interact with simulation interfaces on a daily basis, perceived the simulation as time-consuming (e.g., *"I'm*

---

[10] Quote translated from Danish.

used to work under very high work pressure, so getting in and checking such things through that way is not in my habits" [10], *"You would spend too long to press and read all four options, then press again and read three new options, then press again and there will be five new options"* [10]).

The analysis conducted in this section shows that each artifact has some strengths but also presents some weaknesses, which are summarized in Table 2. Despite having been exposed to the three process artifacts during all the comprehension tasks, participants showed a preference for certain process artifacts based on their domain knowledge, the perceived usefulness and the context in which the artifacts have been deployed. Participants have also reported a set of challenges they faced when interacting with these artifacts.

The perceived benefits and challenges identified by users are often based on their background and working habits and, therefore, are likely to influence the use of the different artifacts. For this reason, benefits and challenges are also suitable to complement the findings of the analysis introduced in Section 5.1 and to provide a deeper explanation of certain use behaviors. For example, from the analysis of the attention maps of Fig. 7a we observed that the simulation was used rarely by municipal employees, especially when compared with the other artifacts. As Table 2 reports, the think-aloud data reveal that municipal employees considered it as "time-consuming" for a person that is not used to deal with such tools in his or her working life. Similarly, the inherent complexity of the legal text seems to have discouraged academics to use it, as reflected in attention map of Fig. 7b.

Overall, the results presented in this section suggest that participants with different backgrounds perceive different benefits from the use of different artifacts. Thereby, we can claim that combining all these artifacts into a hybrid representation makes their use more effective and eases the task of meeting individual user preferences.

5.3 What Strategies Are Followed When Engaging with the Different DCR-HR Artifacts? (RQ3)

By analyzing the data following the procedure outlined in Section 4.3 we were able to classify the behavior of participants into goal-directed and exploratory. We noticed that participants exhibiting a goal-directed behavior fixated mainly the question title and the AOIs in the DCR graph deemed relevant for a particular task. Moreover, we observed that fixations on the law text often occurred when solving decision tasks (i.e., when contextual information is needed) while fixations on the simulation often occurred when solving scenario tasks (i.e., when information about execution traces is required). In addition, we noticed that the relevant AOIs of the graph were either visited following a sequential pattern, that is by visiting one relevant AOI after the other, or following a more fragmented pattern, that is by switching frequently between the question title and the relevant AOIs. As for the participants who exhibited an exploratory behavior, we observed that they fixated more AOIs in the DCR Graph. However, these AOIs were often irrelevant for answering the given tasks. Moreover, we noticed that these participants were continuously intertwining between the different artifacts of DCR-HR and within different parts of the DCR Graph without focusing on a particular AOI, which in turn hints towards a lack of a guided strategy to solve the given tasks.

Fig. 9 illustrates the visual behavior (represented as sequences of fixations projected on the stimulus and scarf-plots) of two representative participants solving a constraint task: participant P09 following a goal-directed strategy and P14 exhibiting an exploratory strategy.

The considered constraint task was asking the following question: "When the municipal council offers 15 hours part-time assistance, then it is mandatory for the receiver to designate a person. True or False?". Clearly, since the question mentions information that is explicitly represented in the DCR Graph, we identified activities *offer 15 hours of assistance* and *designate a person* as relevant, together with the condition connecting them (cf. Fig. 1). In Fig. 4 these correspond to the areas of interest labeled AOI_OFFER_15_HOURS, AOI_DESIGNATE_PERSON and CONDITION_OFFER_DESIGNATE. All the other areas of the graph are considered non-relevant for this specific task, whereas the law text and the simulation could in principle being used to support the user in responding to the question.

As regards to the number of visited AOIs, P09 visited only 5 out of the 22 AOIs defined on the DCR Graph and most of the fixations were on relevant AOIs. In addition, a large portion of these fixations lasted for a considerable period of time: P09 spent the 35.31% of the total fixation time on relevant AOIs, while the same proportion for non-relevant AOIs of the graph amounts to 3.74%. By considering the temporal order of the fixations, it is easy to see that P09 had several long and repeated fixations on the question title at the beginning of the comprehension task. Afterwards, the participant visited AOI_OFFER_15_HOURS and AOI_DESIGNATE_PERSON in sequential order for some time before switching back to the question title.

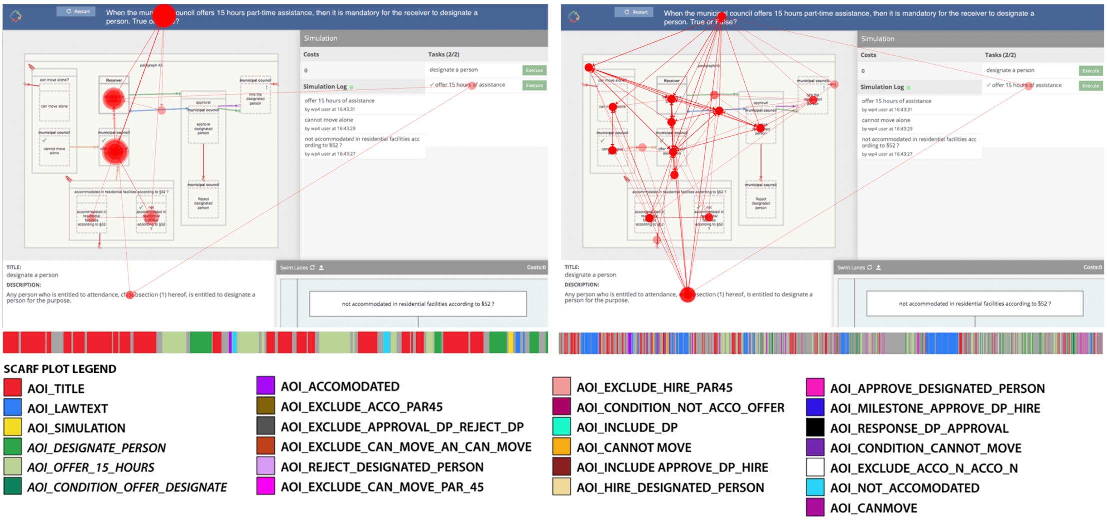


Fig. 9 Comparing the visual behavior of participant P09 following a goal-directed strategy to solve a constraint task (left) to the one of participant P14 following an exploratory strategy (right).

The same pattern re-occurred several times throughout the whole comprehension task. In addition, P09 did not focus on artifacts other than the DCR Graph. This suggests that the user is familiar enough with the semantics of DCR Graph and aware of how to derive the needed information from it. We labeled the strategy followed by P09 as "goal-directed".

By taking a look at the behavior of P14, we can see that this participant visited several AOIs during the execution of the task and, precisely, he focused on 16 out of the 22 AOIs covering the DCR Graph. A large amount of these fixations were on non-relevant AOIs and most of them lasted for a short amount of time. Overall, P14 spent the 29.08% of the total fixation time fixating relevant AOIs in the graph, and the 29.33% of it fixating non-relevant ones. By considering the temporal order in which the fixations occurred, one can see that P14 had several short fixations on the title before moving briefly to the law text and then coming back to the title. Afterwards, the participant kept intertwining between different AOIs apparently without focusing a particular one. This pattern continued throughout the whole comprehension task thus suggesting an exploratory strategy.

Indeed, during the retrospective think-aloud, P14 reported some difficulties with remembering all the different kinds of constraints included in the DCR Graph and he admitted having used the law text to improve his own understanding of the different relationships: *"I did remember from the introduction I was given . . . I did remember a few things but through the time I was forgetting a bit what do they mean to me so I was using the hold over* [the law text] *to understand them to see what they mean"*, *"the logic behind them* [the constraints] *was clear was clear but it's a matter of remembering"*.

The goal-directed and exploratory behaviors illustrated for P09 and P14 can be observed in all participants, regardless of the kind task they have been solving. Fig. 10 shows the scarf-plot of all participants related to the constraint task discussed for P09 and P14. As observed for the latter ones, the difference between the goal-directed and exploratory strategies does not lie in the amount of time spent on the relevant AOIs of the graph which (19,97% vs 24,67%), but is rather noticeable in the amount of time spent on non-relevant AOIs (12,83% vs 27,04%). Moreover, we observed that the total task execution time for participants exhibiting a goal-directed behavior was shorter on average compared to the one of those showing an exploratory behavior: the first group took an average of 44 seconds, while the second group took 02 minutes and 18 seconds on average. During the execution of this task, participant P03 exhibited an unusual behavior, as he fixated mostly the title and had only a couple of glimpses on non-relevant AOIs of the DCR Graph. According to [67], long fixations on a misleading element seem indicate an unclear interaction behavior. When looking into the verbal comments gathered with the think-aloud we discovered that P03 was confused about the graph and used common sense to respond to this question, which explains the unusual strategy.

Fig. 11 and Fig. 12 show the visual behavior of the participants solving a decision task and a scenario task, respectively. Considerations similar to those made for Fig. 10 can be made regarding the categorization of users' strategies into goal-directed and exploratory, keeping in mind that for answering the decision question resulting in the plots of Fig. 11 users had to necessarily refer to contextual information included in the law text, while for answering the question resulting in the plots of Fig. 12 the simulation would have made it easier to remember the process trace.

Regardless of the kind of question being solved, we observed other patterns in the use of different artifacts over time. In particular, we noticed that simulation (coded in yellow in Fig. 10 – Fig. 12) was used at different times during task execution, that is, either immediately after having read the question title (as done by P06 in Fig. 11 or by P05 in Fig. 10 and in Fig. 12) or towards the end of the task, as done by (P02, P06, P11 and P13 in Fig. 12).

This, in turn, suggests two different ways of using simulation: people using it at the beginning of the task may have exploited it as a support for understanding the question and to improve the interpretation of the graph, while people using it towards the end of the task may have used it to validate or confirm a hypothesis they had made by looking at the graph or the law text.

By looking into the think-aloud, we found evidence of these two different uses of the simulation. Participant P06 seems to confirm that simulation is useful to understand what the question is asking, while P11 reports that the simulation eases the comprehension of the different relations in the graph.

More precisely, P06 mentions *"I think I used a lot the simulator. It is useful. Basically, you just base on the question and you follow the question and you do it"*. *"For example in a question there will be kind of assumptions or kind of simulation about if you are alone or with your mom. I used the simulator and I said - Ok, what is the possibility and what are the next steps - then I would say - Ok back to the text* [i.e., the question]*... what he wants me to do? He wants me to go for this possibility, then I click this one - so I can align the text and the simulator."* P11 reports *"I used the simulator and then I learned some things that I wasn't sure about. Also if I was sure about what would happened in one sub-process and then I saw it, then next time I knew how it worked"*.

Other participants confirmed having used the simulation as a way to confirm something they were not sure about or something they had already hypothesized when looking at the graph. Participant P05 claims to have used the simulator as an exclusion criterion: *"Those* [the questions for which he used the simulator] *were the ones where there were more criteria put into the question. Then I used it as a method of exclusion, I think."*. Again, participant P11 reports having used the simulator to confirm the answers he had in mind after having looked at the model *"I use it* [the simulator] *after... if I use the model and I'm not sure or maybe I just want to convince myself. Like, maybe I think this is the right answer but just to make sure I can run the simulator because it kinds of makes me surer."*.

Last but not least, considering the sequential order in which the tasks were displayed to the participants and the strategies adopted when executing different kinds of tasks, we noticed a trend towards switching to a more goal-directed strategy as each eye-tracking session proceeded. Table 3 summarizes the kind of behavior adopted by each participant when solving the constraint task (cf. Fig. 10), a decision task (cf. Fig. 11) and a scenario task (cf. Fig. 12). Hereafter, one could notice that some of the participants who started with an exploratory behavior switched to a goal-directed one during the eye-tracking sessions.

Participant P11 provided a possible explanation for this finding in the think-aloud: *"The first round I spent a lot of time looking at the whole model even though it didn't have anything to do with the questions because I wasn't sure if I just missed something and then I could use that information later on."*. This kind of behavior seems to find confirmation in the fact that exploratory search can sometimes operate as a screening process that identifies candidates for goal-directed search [24].

6 Discussion

In this section, we discuss the findings of this study in light of existing literature. Sections 6.1, 6.2 and 6.3 discuss the findings associated with **RQ1**, **RQ2**, and **RQ3** respectively, while Section 6.4 discusses the limitations of this work.

6.1 How Do Users Engage with the Different DCR-HR Artifacts? (RQ1)

The results presented in Section 5.1 suggest that people interact differently with the artifacts of DCR-HR based on their background and the kind of task they are executing. The way different backgrounds affect the use of DCR-HR is in line with the claims made by previous researchers [2, 3, 44] to support the purposeful perspectives offered by hybrid process artifacts. Moreover, our findings reflect the circumstance where

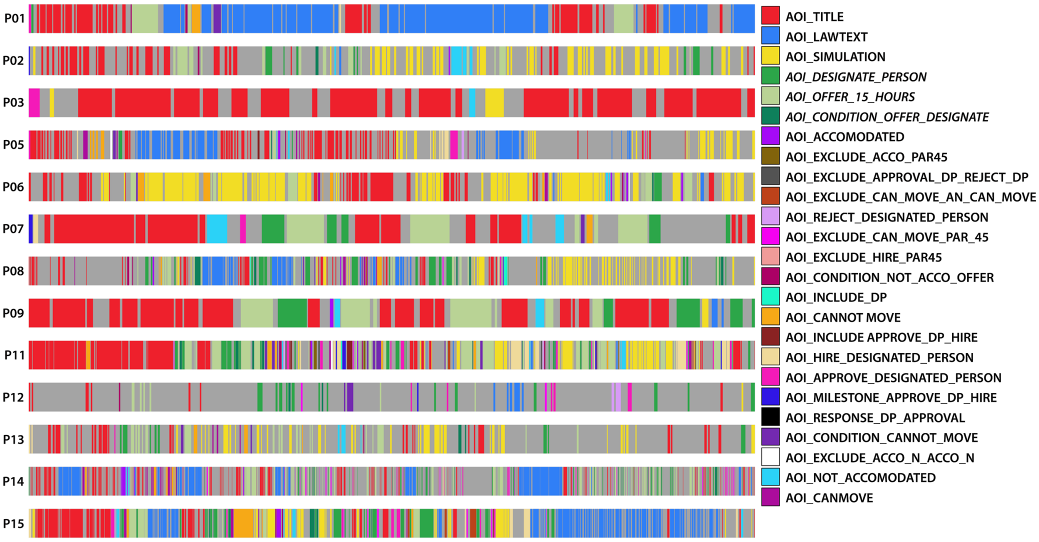


Fig. 10 Scarf-plot showing the sequences of fixations for participants solving a constraint task and classified into goal-directed (GB) and exploratory (EB) strategies. Relevant AOIs of the DCR Graph for this task are labeled in italic.

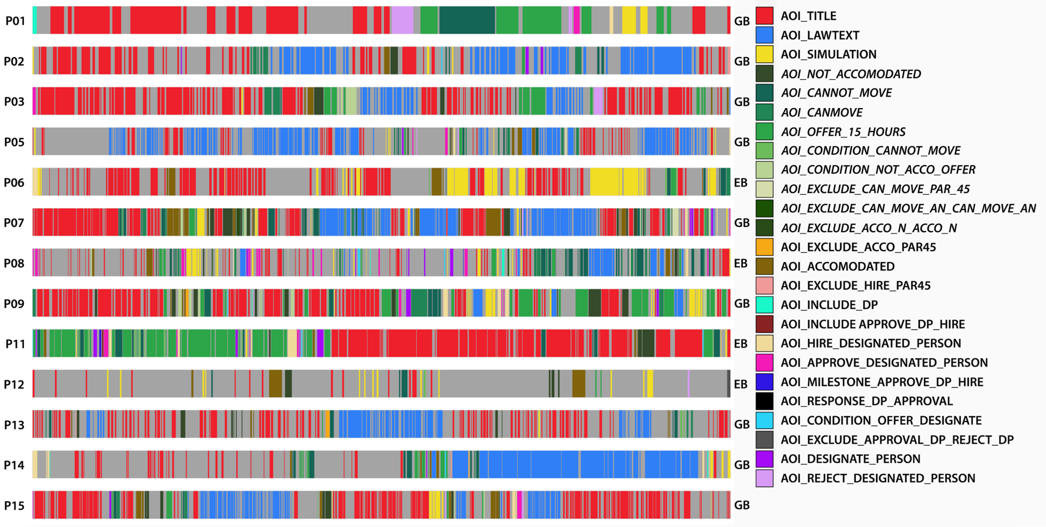


Fig. 11 Scarf-plot showing the sequences of fixations for participants solving a decision task and classified into goal-directed (GB) and exploratory (EB) strategies. Relevant AOIs of the DCR Graph for this task are labeled in italic.

both domain experts (represented by municipal employees) and IT specialists (represented by academics) are challenged when being exposed to unfamiliar process artifacts [3]. The deployment of a hybrid process artifact can help to overcome this issue by providing a representation that is comprehensible to both stakeholders.

Existing research associates the comprehension of business processes with the effectiveness of communication between different stakeholders [3,68]. Using different levels of formality (i.e., natural language and DCR notation) and different levels of abstraction (i.e., a process model abstraction and an instance-based simulation) hybrid process artifacts could foster the communication between different groups of stakeholders by providing the means to clarify the terms and relationships in the domain and prevent misinterpretations.

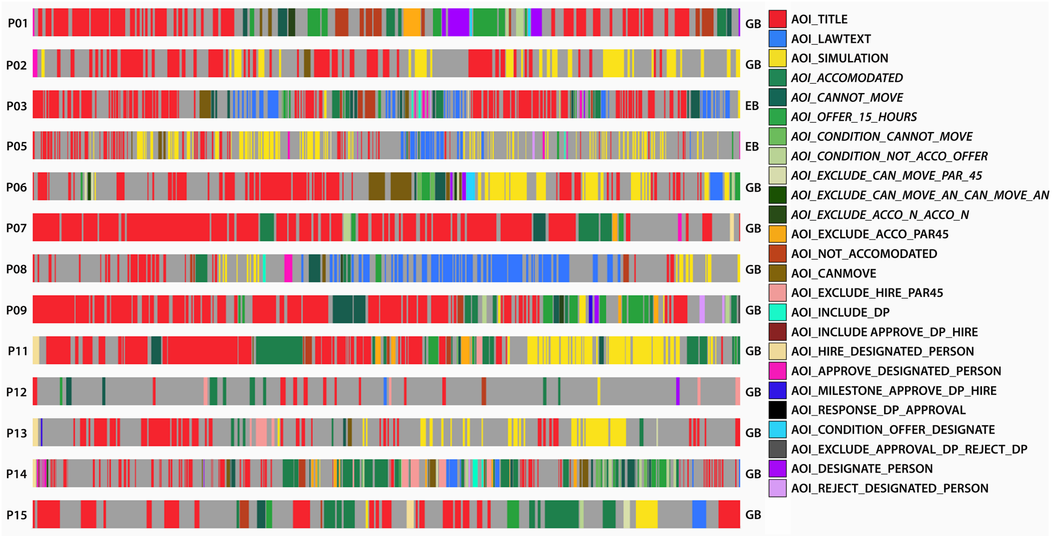


Fig. 12 Scarf-plot showing the sequences of fixations for participants solving a scenario task and classified into goal-directed (GB) and exploratory (EB) strategies. Relevant AOIs of the DCR Graph for this task are labeled in italic.

													TOTAL	
Task type	P01	P02	P03	P05	P06	P07	P08	P09	P11	P12	P13	P14	P15	**GB**	**EB**
Constraint	GB	GB	–	EB	GB	GB	EB	GB	EB	GB	GB	EB	EB	**7**	**5**
Decision	GB	GB	GB	GB	EB	GB	EB	GB	EB	EB	GB	GB	GB	**9**	**4**
Scenario	GB	GB	EB	EB	GB	GB	GB	GB	GB	GB	GB	GB	GB	**11**	**2**

Table 3 Summary of goal-directed and exploratory strategies across tasks. GB refers to Goal-directed Behavior; EB refers to Exploratory Behavior.

The disparity of task types is clearly reflected in the reading patterns of the participants. From the analysis conducted in Section 5.2, it has emerged that the participants have changed their reading patterns according to the task they were executing. This suggests that different combinations of artifacts have been used in a context-specific manner to address different comprehension tasks.

The relation between task types and the underlying cognitive processes are discussed in the literature in a bi-directional fashion. In cognitive psychology, the impact of the task type on the underlying cognitive processes is discussed by Glaholt et al. [69]. Following a quantitative approach, the authors investigate the eye gaze selectively (i.e., the extent to which an individual is selective when being asked to choose between different alternatives) associated with different types of tasks. They discovered that the task type has a direct influence on eye-tracking measures, such as the total fixation duration, number of dwells, and their mean duration. Grounded on a set of visual patterns, our attention maps come to support Glaholt's insights from a qualitative perspective showing the difference in users' reading patterns when approaching different model comprehension tasks.

In the field of process modeling, existing research shows the potential of deploying users' behavioral patterns to predict the task they are involved in (i.e., problem understanding, method finding, validation) [70]. Interestingly, many of the measures deployed in [70] (i.e., dwells and transitions between AOIs) to provide accurate predictions about the task at hand, have also been used in the present study as a basis to generate the attention maps showing the distinctive reading patterns. Hence, the emerging distinction of task types could be operationalized using the underlying measures to provide context-adaptive tool-support at run-time, making relevant aspects more salient and accessible for the user.

6.2 What Are the Benefits and Challenges Associated with Each One of the Artifacts of DCR-HR? (RQ2)

The retrospective think-aloud provided deeper insights into the way users engaged with the different DCR-HR artifacts. These subjective insights provide a means to explain the different reading patterns inferred from the eye-tracking data – but also to identify the challenges associated with the different DCR-HR artifacts. The challenges associated with the comprehension of the DCR Graphs intersect to a large extent with the notorious challenges of declarative process models [5, 28–30]. When it comes to the preference for textual languages (e.g., law text) over graphical ones (e.g., DCR Graphs) or the other way around, the background of the participants seems to play a central role. A similar finding is reported by Ottensooser et al. [71]. The authors associate the experience of the user with a particular language to the effectiveness of its use. This is also supported by other literature (e.g., [72, 73]) emphasizing the importance of individuals' familiarity with the language being used and its impact on creating a better cognitive fit between the given material and the task in hand. As a result, this insight could explain the preference of municipal employees to law text and academics to DCR-graphs. Besides this personal factor inherent to the participants' background, the existing literature remains divided on whether graphical languages are more comprehensible than textual ones. Throughout different studies, authors compare several textual and graphical languages, and the results are split based on the languages being compared [71, 74, 75]. One the one hand, graphical languages for process modeling can be efficient with regards to their spatial arrangement of the modeling constructs, allowing the reader to perceive the interplay between the different constructs easily [76]. However, the deployed graphical notations are not as intuitive as they claim to be [77] and usually require formal training to be understood [78]. On the other hand, textual languages can be readily comprehended and interpreted by non-experts [71,78]. The exploratory insights of this study suggest that this argument could only hold for readers having a precedent textual aptitude for a specific domain terminology; otherwise, the deployed vocabulary would still remain challenging to be understood. This is exactly the case of academics, who were mostly challenged by reading the law text. The deployment of a hybrid process artifact could, in turn, help overcoming the challenges associated with the individual artifacts. In cognitive psychology, this suggestion is largely supported by the dual coding theory [79], which highlights the importance of combining textual and graphical artifacts to reinforce the understandability of the material in hand.

Looking further into the subjective insights provided by the participants, it seems that academics and municipal employees perceived different precision and rigidity in the DCR Graph and the law text. These perceptions can be seen as being subjective to the participants' background. Indeed, academics did not feel comfortable with the fuzzy law text and showed a preference for the process model, being a formal artifact prescribing the exact behavior of the model. In contrast, municipal employees rather used the law text, which is less formal and leaves room for diverse interpretations. While it is true that a formal model is more precise and less ambiguous than law text, it might not be sufficient for solving all possible cases. As the level of flexibility supported by the model is predetermined at design time, a model could either encode a strict interpretation of the law, or a loose one, but never both. The law text, in contrast, can support both flexible and strict interpretations depending on the way the reader perceives it. Another possible limitation of the model is due to its inability to specify the most desired path and outcome of the process, whereas the law text could explicitly express this requirement.

6.3 What Strategies Are Followed When Engaging with the Different DCR-HR Artifacts? (RQ3)

Among the findings of this study is the mapping between the different strategies enacted by participants and the visual search behaviors (i.e., goal-directed, exploratory) introduced in the literature (cf. Section 2.3). Metrics such as the distribution of fixations over time, their frequency and duration, and the number of fixated AOIs were previously used in quantitative studies to distinguish different types of visual search behaviors [25] or to identify usability issues [67]. In this paper, we followed a qualitative approach supported by eye-tracking to map the strategies enacted by participants to the existing visual search behaviors and backed up the interpretation of our data with insights derived from the think-aloud.

The time spent fixating non-relevant AOIs was among the key features used for this classification. According to [46] the time spent fixating relevant regions of a process model can predict answer correctness. In our study, we did not notice such correlation between the time spent on relevant AOIs and the amount of correct answers. However, while the authors in [46] consider BPMN process models and focus on structure understanding, we focus on general comprehension questions on DCR Graphs, whose declarative nature requires people to comprehend the different constraints involved and hidden dependencies [53]. In this context, even if a relevant AOI is fixated for a long time, a misunderstanding of the constraints semantics may compromise the correctness of the provided answers. Moreover, especially for decision tasks, looking solely at relevant AOIs in the process model was not enough to come up with a correct answer.

In usability studies, short fixations targeting a large number of non-relevant AOIs have been associated to confusion generated from the inability of the user to find a certain piece of information [67]. In the think-aloud, confusion was indeed one among the most recurring issues reported by participants, mostly in relation with the DCR Graph or the law text. Yet, some people claimed that their understanding of the information conveyed by the three artifacts improved throughout the study, thus allowing them to make a more effective use of each artifact and resolving part of the initial confusion. This behavior could explain why we noticed a change from exploratory to goal-directed strategies over time. Such change of behavior across eye-tracking sessions is also reflected in the findings of a recent experiment conducted by Zhao et al. [39] in the context of text-picture comprehension. According to the authors, pictures are likely to be used for task-oriented specific processing, but participants require some time to construct their own mental model before using them. Although DCR Graphs are not simple pictures as they include textual labels and a graphical notation having a precise semantics, we found evidence of such behavior in the think-aloud data (cf. Section 5.3) and, therefore, we may consider the findings in [39] as a possible explanation for the change in strategies.

Last but not least, we were able to observe also other patterns describing the use of the different DCR-HR artifacts and, specifically, the use of simulation. Declarative process models such as DCR Graphs are known to be well-suited to convey circumstantial information, while sequential information (e.g., traces) remains implicit [29]. Thus, to determine whether a particular trace is supported or not, or to determine possible following actions based on a partial trace, the user is required to keep numerous states in mind. The simulation, in turn, makes the sequential information explicit and allows to offload memory facilitating the execution of scenario tasks. In this study, we noticed that some participants used simulation starting from the initial phases of the comprehension task as a means to improve their understanding of the question or of the DCR Graph. Instead, others used the simulation towards the end of the comprehension task, to sort of validate their hypothesized answers. Pinggera highlights a similar use of validation during process modeling. In his PhD work [55], he reports about the difference between incremental and final validation. In the first case, validation, co–occurs with the modeling and reconciliation phases, indicating that the modeler spent some time evaluating the designed process model. In the second case and most common scenario, validation is observed mostly at the end and the time spent validating the model seems to increase according to the size and the complexity of the model.

6.4 Threats to Validity

There are number of threats associated with the validity of our research. In the following we discuss these threats and evoke the actions taken to mitigate their effects.

Internal Validity. The design of the study is subject to some threats. Our study uses a relatively small sample size, which is nonetheless acceptable in exploratory studies [80, 81]. Moreover, disparities in the expertise of the participants might have limited the interactions of the least expert participants with DCR-HR and caused them to overlook some of the features of the DCR Portal. To mitigate this effect, all participants were taught the semantics of the DCR notation and were uniformly familiarized with the main features of the DCR Graphs Portal. Furthermore, participants with different levels of expertise might have exhibited slightly different behaviors compared to other participants in the same group. However, these differences did not clearly emerge during the analysis. The interactions between the researchers and participants is another possible threat to validity. To avoid this risk, a data collection protocol specifying all the steps of the data collection procedure has been followed during all sessions guaranteeing that all participants receive the same instructions and ensuring that the researcher is not biasing the insights provided by the participants.

External Validity. The design of the study, its exploratory nature, and the limited number of participants make the findings difficult to generalize. Nevertheless, the obtained insights allowed the identification of different reading patterns and strategies, which could be used as a basis for future investigations. In addition, the covered task types represent only a small subset of all the possible circumstances where process models are used in the real-world. Although the choice of these tasks was motivated by the input of experts in the field of process modeling, with close ties to legal practitioners and municipal employees, their elicitation was driven by practical motivations rather than theoretical foundations, which could be considered in future studies.

We could report two main strategies describing the behavior of people engaging DCR-HR and observed that people tend to become more goal-directed while becoming more acquainted with different artifacts. Although we were able to categorize the behavior of people into goal-directed and exploratory, we do not exclude that there are other strategies followed by people when engaging with hybrid process artifacts, such as those outlined in [39] considering a mix of goal-directed and exploratory behaviors depending on when and how comprehension tasks are shown to participants.

The hybrid process artifact we investigate in this work is based on DCR Graphs. Hence, the validity of our insights is bound to that specific language. However, the constraint-based approach applied in DCR is shared with many other declarative languages (e.g., Declare [15]). Therefore, our findings could presumably apply to other languages in the declarative paradigm as IT specialists would probably, like in our study, struggle reading the law text and domain experts would have a preference for the law text. It is however unclear whether our results can be generalized to imperative process models. An imperative model (e.g., in BPMN [47]), would have been possibly easier to understand for domain experts, and thus more interactions with the process model could have occurred.

In our exploratory study the hybrid artifacts were overlapping in terms of the information that each artifact provided. Indeed the simulation and the DCR Graph were information equivalent and the model reflected largely the specifications in the law text. It is rather unlikely that our findings would generalize to hybrid representations with limited information overlap between artifacts (e.g., imperative models enriched with business rules [50]). In this case it might be difficult for domain experts or IT specialists to rely on a single artifact to solve a particular task, but it might become necessary to understand all the artifacts. In such settings it becomes even more important to focus on the quality of the artifacts composing a hybrid representation, since it needs to be ensured that both domain experts and IT specialists can make sense of them.

Construct Validity. The measures used to answers our research questions (cf. Table 1) show differences between groups of participants when dealing with different tasks. The interpretation of these differences can cause a threat to validity if not correctly triangulated with other data sources and supported by existing literature. To reduce this threat, the interpretation of the eye-tracking measures was supported by the participants' verbal utterances and the notes collected from their eye gaze recordings before being linked to the existing body of knowledge in literature.

Reliability. Reproducibility is a crucial requirement for any empirical research. Although our findings are qualitative and use the subjective insights obtained from the participants, we have followed a systematic approach based on concepts from grounded theory (cf. Section 4.3) in order to identify the pertinent aspects evoked by the participants. Nevertheless, the coding procedure might entail some subjectivity which could have biased our findings. To reduce this effect, our coding was constantly reviewed and discussed by the co-authors, ensuring a consensus in coding borderline cases.

7 Conclusion and Future Work

This study takes an initial stride towards providing an in-depth understanding of the way declarative process models are read when combined with other artifacts. Looking at the way participants with different backgrounds engaged with DCR-HR during the execution of different tasks, we observed that municipal employees and academics used artifacts in a different way. Accordingly, the background of the users seems to influence the perceived benefits and challenges associated to each artifact, thus influencing the way they are used. Exploring how different artifacts are used over time, we noticed that people follow different strategies when engaging with DCR-HR and tend switch from an exploratory to a goal-directed behavior as time goes by.

The findings delineate clear directions for future work. First and foremost, it is necessary to investigate the different reading patterns and strategies identified throughout this study in light of the support they provide for a better understanding of the business process. To this end, performance metrics, such as answer accuracy and response time, could be correlated with the identified patterns to discern the most efficient ones. With the availability of more data, the association between reading patterns and understandability and its operationalization through performance metrics could be used in practice to develop a statistical model that could be trained to predict the performance of users based on the patterns they exhibit during a comprehension task.

Another relevant direction for future work is the exploitation of the identified reading patterns and strategies to provide better tool-support for users at run-time. Indeed, the behavioral features of the participants could be used at run-time to determine their background and expertise, and accordingly adjust the hybrid representation by highlighting the relevant areas on the artifacts they are most likely to use. This feature is expected to reduce the cognitive distraction, and the attention split [82] caused by the display of different artifacts. The identification of the task at hand could further reduce such a cognitive effect by putting more emphasis on the artifacts relevant for solving a particular task and shading the irrelevant ones. Automating the identification of the users' strategies at run-time could also contribute to an increased tool support. Herein, additional guidance could be provided to users with exploratory behavior by providing cues allowing to reduce the search area and thus promoting a rather goal-directed behavior.

Understanding the user behavior during a comprehension task can be extended beyond the proxy tasks covered by this study. Such an extension could be done by adopting the proposed qualitative approach in the analysis of the modeling and the maintaining of process models with the support of hybrid process artifacts. Herein, more robust analysis techniques supported by eye-tracking and other biosensor devices could be used to obtain fine-grained insights about the support of hybrid process artifacts, not only with regards to comprehension tasks but also when modeling and maintaining processes.

Besides, our exploratory insights might improve the design and the modeling of hybrid process artifacts. As the background of users influence their preferences for artifacts, it is important to consider the audience targeted by the hybrid representation and pinpoint the less clear aspects in the model requiring to be enriched with additional artifacts. In this vein, future work could investigate the ways in which process artifacts can overlap and propose clear guidelines on designing hybrid process artifacts. Moreover, our findings show that users tend to switch from an exploratory behavior to a goal-directed behavior progressively. Related to that, the quality of process artifacts could be further investigated, ensuring that novices can rapidly make sense of them and thus facilitating the transition from an exploratory behavior to a goal-directed behavior. This is particularly the case for the DCR Graph, which could be modeled in different ways. While many guidelines have emerged to prescribe the factors affecting the quality of imperative process models, little is known about the quality of declarative models. Future work could take this direction to investigate and infer the quality factors affecting the understandability of declarative models, particularly those represented in DCR Graphs.

Overall, the insights arising from this exploratory study are expected to have an important impact on current research. Indeed, the different reading patterns raise several questions about their influence on the comprehension of hybrid process artifacts and the underlying human cognitive processes. This paves the path for future investigations aimed to improve the design of these artifacts. Additionally, steering the direction for future work towards the development of adaptive tool-support, learning and adjusting to the users' behavior, would certainly help to bring the current research to practice by providing run-time support to users.

References

1. R Hevner Von Alan, Salvatore T March, Jinsoo Park, and Sudha Ram. Design science in information systems research. *MIS quarterly*, 28(1):75–105, 2004.
2. Frank Dengler and Denny Vrandečić. Wiki-Based Maturing of Process Descriptions. pages 313–328. 2011.
3. Jakob Pinggera, Thomas Porcham, Stefan Zugal, and Barbara Weber. LiProMo-Literate process modeling. *CEUR Workshop Proceedings*, 855:163–170, 2012.
4. Stefan Zugal, Jakob Pinggera, and Barbara Weber. Creating Declarative Process Models Using Test Driven Modeling Suite. In Selmin Nurcan, editor, *IS Olympics: Information Systems in a Diverse World*, pages 16–32. Springer Berlin Heidelberg, 2012.
5. Stefan Zugal, Jakob Pinggera, and Barbara Weber. Toward enhanced life-cycle support for declarative processes. *Journal of Software: Evolution and Process*, 24(3):285–302, apr 2012.
6. Johannes De Smedt, Jochen De Weerdt, Estefanía Serral, and Jan Vanthienen. Improving Understandability of Declarative Process Models by Revealing Hidden Dependencies. In Selmin Nurcan, Pnina Soffer, Marko Bajec, and Johann Eder, editors, *Advanced Information Systems Engineering*, pages 83–98, Cham, 2016. Springer International Publishing.
7. Johannes De Smedt, Jochen De Weerdt, Estefanía Serral, and Jan Vanthienen. Discovering hidden dependencies in constraint-based declarative process models for improving understandability. *Information Systems*, 74:40–52, may 2018.
8. Hugo A López, Søren Debois, Thomas T Hildebrandt, and Morten Marquard. The process highlighter: From texts to declarative processes and back. In *BPM (Dissertation/Demos/Industry)*, pages 66–70, 2018.
9. Amine Abbad Andaloussi, Jon Buch-Lorentsen, Hugo A. López, Tijs Slaats, and Barbara Weber. Exploring the modeling of declarative processes using a hybrid approach. In Alberto H. F. Laender, Barbara Pernici, Ee-Peng Lim, and José Palazzo M. de Oliveira, editors, *Conceptual Modeling*, pages 162–170, Cham, 2019. Springer International Publishing.
10. Amine Abbad Andaloussi, Andrea Burattin, Tijs Slaats, Anette Chelina Møller Petersen, Thomas T. Hildebrandt, and Barbara Weber. Exploring the understandability of a hybrid process design artifact based on dcr graphs. In Iris Reinhartz-Berger, Jelena Zdravkovic, Jens Gulden, and Rainer Schmidt, editors, *Enterprise, Business-Process and Information Systems Modeling*, pages 69–84, Cham, 2019. Springer International Publishing.
11. Amine Abbad Andaloussi, Andrea Burattin, Tijs Slaats, Ekkart Kindler, and Barbara Weber. On the declarative paradigm in hybrid business process representations: A

conceptual framework and a systematic literature study. *Information Systems*, 91:101505, 2020.

12. Manfred Reichert and Barbara Weber. *Enabling Flexibility in Process-Aware Information Systems*. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.
13. Stefan Zugal, Jakob Pinggera, and Barbara Weber. The impact of testcases on the maintainability of declarative process models. In Terry Halpin, Selmin Nurcan, John Krogstie, Pnina Soffer, Erik Proper, Rainer Schmidt, and Ilia Bider, editors, *Enterprise, Business-Process and Information Systems Modeling*, pages 163–177, Berlin, Heidelberg, 2011. Springer Berlin Heidelberg.
14. Johannes De Smedt, Jochen De Weerdt, Jan Vanthienen, and Geert Poels. Mixed-paradigm process modeling with intertwined state spaces. *Business and Information Systems Engineering*, 58(1):19–29, 2016.
15. Maja Pesic, Helen Schonenberg, and Wil M.P. van der Aalst. DECLARE: Full Support for Loosely-Structured Processes. In *11th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007)*, pages 287–287. IEEE, oct 2007.
16. Søren Debois, Thomas Hildebrandt, Morten Marquard, and Tijs Slaats. Hybrid Process Technologies in the Financial Sector. *Proceedings of the Industry Track at the 13th International Conference on Business Process Management 2015*, 4:107–119, 2015.
17. Thomas T. Hildebrandt and Raghava Rao Mukkamala. Declarative Event-Based Workflow as Distributed Dynamic Condition Response Graphs. *Electronic Proceedings in Theoretical Computer Science*, 69:59–73, 2011.
18. Morten Marquard, Muhammad Shahzad, and Tijs Slaats. Web-Based Modelling and Collaborative Simulation of Declarative Processes. In Hamid Reza Motahari-Nezhad, Jan Recker, and Matthias Weidlich, editors, *Business Process Management*, pages 209–225, Cham, 2016. Springer International Publishing.
19. K. Holmqvist, M. Nyström, R. Andersson, R. Dewhurst, H. Jarodzka, and J. van de Weijer. *Eye Tracking: A comprehensive guide to methods and measures*. OUP Oxford, 2011.
20. Jennifer Romano Bergstrom and Andreei Schall. *Eye tracking in user experience design*. Elsevier, 2014.
21. Georg Buscher, Andreas Dengel, Ralf Biedert, and Ludger V Elst. Attentive documents: Eye tracking as implicit feedback for information retrieval and beyond. *ACM Transactions on Interactive Intelligent Systems (TiiS)*, 1(2):9, 2012.
22. Wil M.P. Van Der Aalst. *Process mining: Discovery, Conformance, and Enhancement of Business processes*. Springer International Publishing, 2010.
23. Tanja Blascheck, Kuno Kurzhals, Michael Raschke, Michael Burch, Daniel Weiskopf, and Thomas Ertl. Visualization of eye tracking data: A taxonomy and survey. In *Computer Graphics Forum*, volume 36, pages 260–284. Wiley Online Library, 2017.
24. Chris Janiszewski. The Influence of Display Characteristics on Visual Exploratory Search Behavior. *Journal of Consumer Research*, 25(3):290–301, 12 1998.
25. Jella Pfeiffer, Martin Meißner, Jascha Prosiegel, and Thies Pfeiffer. Classification of goal-directed search and exploratory search using mobile eye-tracking. *Proceedings of the International Conference on Information Systems*, 2014.
26. W. Maalej, M. Nayebi, T. Johann, and G. Ruhe. Toward data-driven requirements engineering. *IEEE Software*, 33(1):48–54, Jan 2016.
27. Tijs Slaats. Declarative and hybrid process discovery: Recent advances and open challenges. *Journal on Data Semantics*, pages 1–18, 2020.
28. M. Pesic. *Constraint-based workflow management systems : shifting control to users*. PhD thesis, TU Eindhoven, 2008.
29. Dirk Fahland, Daniel Lübke, Jan Mendling, Hajo Reijers, Barbara Weber, Matthias Weidlich, and Stefan Zugal. Declarative versus Imperative Process Modeling Languages: The Issue of Understandability. In *Proceedings of EMMSAD*, pages 353–366. Springer, 2009.
30. Dirk Fahland, Jan Mendling, Hajo A Reijers, Barbara Weber, Matthias Weidlich, and Stefan Zugal. Declarative versus Imperative Process Modeling Languages: The Issue of Maintainability. In Stefanie Rinderle-Ma, Shazia Sadiq, and Frank Leymann, editors, *Business Process Management Workshops*, pages 477–488, Berlin, Heidelberg, 2010. Springer Berlin Heidelberg.
31. George A Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81, 1956.
32. Søren Debois, Thomas Hildebrandt, Morten Marquard, and Tijs Slaats. Bridging the valley of death: A success story on danish funding schemes paving a path from technology readiness level 1 to 9. In *Proceedings of the Second International Workshop on Software Engineering Research and Industrial Practice*, pages 54–57, Piscataway, NJ, USA, 2015. IEEE Press.
33. Tijs Slaats. *Flexible Process Notations for Cross-organizational Case Management Systems*. PhD thesis, IT University of Copenhagen, January 2015.
34. Søren Debois, Thomas T Hildebrandt, Morten Marquard, and Tijs Slaats. The dcr graphs process portal. In *BPM (Demos)*, pages 7–11, 2016.
35. Kuno Kurzhals, Brian Fisher, Michael Burch, and Daniel Weiskopf. Eye tracking evaluation of visual analytics. *Information Visualization*, 15(4):340–358, 2016.
36. Kenneth Holmqvist, Marcus Nyström, and Fiona Mulvey. Eye tracker data quality: what it is and how to measure it. In *Proceedings of the symposium on eye tracking research and applications*, pages 45–52. ACM, 2012.
37. Jens Gulden, Andrea Burattin, Amine A. Andaloussi, and Barbara Weber. From analytical purposes to data visualizations: a decision process guided by a conceptual framework and eye tracking. *Software & Systems Modeling*, Jul 2019.
38. Kuno Kurzhals, Brian Fisher, Michael Burch, and Daniel Weiskopf. Evaluating visual analytics with eye tracking. In *Proceedings of the Fifth Workshop on Beyond Time and Errors: Novel Evaluation Methods for Visualization*, BELIV '14, pages 61–69, New York, NY, USA, 2014. ACM.
39. Fang Zhao, Wolfgang Schnotz, Inga Wagner, and Robert Gaschler. Eye tracking indicators of reading approaches in text-picture comprehension. *Frontline Learning Research*, 2(5):46–66, 2014.
40. Stefan Zugal. *Applying Cognitive Psychology for Improving the Creation, Understanding and Maintenance of Business Process Models*. PhD thesis, University of Innsbruck, 2013.
41. Kathrin Figl. Comprehension of procedural visual business process models. *Business & Information Systems Engineering*, 59(1):41–67, Feb 2017.
42. F. Garcia, F. Ruiz, and C. A. Visaggio. A proposal and empirical validation of metrics to evaluate the maintainability of software process models. In *2006 IEEE Instrumentation and Measurement Technology Conference Proceedings*, pages 1093–1097, April 2006.

43. Jan Claes, Irene Vanderfeesten, Frederik Gailly, Paul Grefen, and Geert Poels. The structured process modeling theory (spmt) a cognitive view on why and how modelers benefit from structuring the process of process modeling. *Information Systems Frontiers*, 17(6):1401–1425, Dec 2015.
44. Stefan Zugal, Cornelia Haisjackl, Jakob Pinggera, and Barbara Weber. Empirical evaluation of test driven modeling. *International Journal of Information System Modeling and Design*, 4(2):23–43, April 2013.
45. Andrew T. Duchowski. *Eye Tracking Methodology: Theory and Practice.* Springer-Verlag, Berlin, Heidelberg, 2007.
46. Razvan Petrusel and Jan Mendling. Eye-tracking the factors of process model comprehension tasks. In Camille Salinesi, Moira C. Norrie, and Óscar Pastor, editors, *Advanced Information Systems Engineering*, pages 224–239, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.
47. Object Management Group OMG. Business Process Modeling Notation V 2.0. 2006.
48. Palash Bera, Pnina Soffer, and Jeffrey Parsons. Using eye tracking to expose cognitive processes in understanding conceptual models. *MIS Quarterly: Management Information Systems*, 43:1105–1126, 2019.
49. G Keller, M Nüttgens, and A W Scheer. *Semantische Prozessmodellierung auf der Grundlage "ereignisgesteuerter Prozessketten (EPK)".* Institut für Wirtschaftsinformatik Saarbrücken: Veröffentlichungen des Instituts für Wirtschaftsinformatik. Inst. für Wirtschaftsinformatik, 1992.
50. Wei Wang, Marta Indulska, Shazia Sadiq, and Barbara Weber. Effect of linked rules on business process model understanding. In *Business Process Management*, pages 200–215. Springer International Publishing, Cham, 2017.
51. B. Sapkota and M. v. Sinderen. Exploiting rules and processes for increasing flexibility in service composition. In *2010 14th IEEE International Enterprise Distributed Object Computing Conference Workshops*, pages 177–185, Oct 2010.
52. Michael Zimoch, Rüdiger Pryss, Johannes Schobel, and Manfred Reichert. Eye tracking experiments on process model comprehension: Lessons learned. In Iris Reinhartz-Berger, Jens Gulden, Selmin Nurcan, Wided Guédria, and Palash Bera, editors, *Enterprise, Business-Process and Information Systems Modeling*, pages 153–168, Cham, 2017. Springer International Publishing.
53. Cornelia Haisjackl, Irene Barba, Stefan Zugal, Pnina Soffer, Irit Hadar, Manfred Reichert, Jakob Pinggera, and Barbara Weber. Understanding declare models: strategies, pitfalls, empirical results. *Software and Systems Modeling*, 15(2):325–352, 2016.
54. Cornelia Haisjackl, Pnina Soffer, Shao Yi Lim, and Barbara Weber. How do humans inspect BPMN models: an exploratory study. *Software and Systems Modeling*, 17(2):655–673, 2018.
55. Jakob Pinggera. *The Process of Process Modeling.* PhD thesis, Department of Computer Science, 2014.
56. Jakob Pinggera, Pnina Soffer, Dirk Fahland, Matthias Weidlich, Stefan Zugal, Barbara Weber, Hajo A Reijers, and Jan Mendling. Styles in business process modeling: an exploration and a model. *Software & Systems Modeling*, 14(3):1055–1080, 2015.
57. Victor R Basili Gianluigi Caldiera and H Dieter Rombach. The goal question metric approach. *Encyclopedia of software engineering*, pages 528–532, 1994.
58. K. Charmaz. *Constructing Grounded Theory.* Introducing Qualitative Methods series. SAGE Publications, 2014.
59. Daniel C Richardson and Rick Dale. Looking to understand: The coupling between speakers' and listeners' eye movements and its relationship to discourse comprehension. *Cognitive science*, 29(6):1045–1060, 2005.
60. A Olsen. The tobii i-vt fixation filter. *Tobii Technology*, 2012.
61. IEEE Standard for eXtensible Event Stream (XES) for Achieving Interoperability in Event Logs and Event Streams. *IEEE Std 1849-2016*, pages 1–50, Nov 2016.
62. A. J. M. M. Weijters, Wil M P van der Aalst, and A. K. Alves De Medeiros. Process Mining with the Heuristics Miner Algorithm. *TU/e Tech. Report*, 166:1–34, 2006.
63. Kevin R. Coombes, Guy Brock, and Zachary B. Abrams. Polychrome: Creating and assessing qualitative palettes with many colors. *Journal of Statistical Software*, 2019.
64. Jens Gulden. Visually comparing process dynamics with rhythm-eye views. In *BPM Workshops*, LNBIP, pages 474–485. Springer, 2016.
65. J. Saldana. *The Coding Manual for Qualitative Researchers.* SAGE Publications, 2012.
66. Susanne Friese. *Qualitative data analysis with ATLAS. ti.* SAGE Publications Limited, 2019.
67. Claudia Ehmke and Stephanie Wilson. Identifying web usability problems from eye-tracking data. In *Proceedings of the 21st British HCI Group Annual Conference on People and Computers: HCI... but not as we know it-Volume 1*, pages 119–128. British Computer Society, 2007.
68. Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo A. Reijers. *Fundamentals of Business Process Management.* Springer Berlin Heidelberg, Berlin, Heidelberg, 2018.
69. Mackenzie G. Glaholt, Mei-Chun Wu, and Eyal M. Reingold. Evidence for top-down control of eye movements during visual decision making. *Journal of Vision*, 10(5):15–15, 05 2010.
70. Andrea Burattin, Michael Kaiser, Manuel Neurauter, and Barbara Weber. Learning process modeling phases from modeling interactions and eye tracking data. *Data and Knowledge Engineering*, 121:1 – 17, 2019.
71. Avner Ottensooser, Alan Fekete, Hajo A. Reijers, Jan Mendling, and Con Menictas. Making sense of business process descriptions: An experimental comparison of graphical and textual notations. *Journal of Systems and Software*, 85(3):596 – 606, 2012. Novel approaches in the design and implementation of systems/software architecture.
72. B. Curtis, S. B. Sheppard, E. Kruesi-Bailey, J. Bailey, and D. A. Boehm-Davis. Experimental evaluation of software documentation formats. *J. Syst. Softw.*, 9(2):167–207, February 1989.
73. Wolff-Michael Roth and Gervase Michael Bowen. When are graphs worth ten thousand words? an expert-expert study. *Cognition and Instruction*, 21(4):429–473, 2003.
74. Thomas G. Moher, Dr. K. H. Mak, B. B. Blumenthal, and L. M. Leventhal. Comparing the comprehensibility of textual and graphical programs. 1993.
75. Cornelia Haisjackl and Stefan Zugal. Investigating differences between graphical and textual declarative process models. In Lazaros Iliadis, Michael Papazoglou, and Klaus Pohl, editors, *Advanced Information Systems Engineering Workshops*, pages 194–206, Cham, 2014. Springer International Publishing.
76. Jill H. Larkin and Herbert A. Simon. Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11(1):65 – 100, 1987.

77. Daniel Moody. The “physics” of notations: Toward a scientific basis for constructing visual notations in software engineering. *IEEE Trans. Softw. Eng.*, 35(6):756–779, November 2009.
78. Keng Siau. Informational and computational equivalence in comparing information modeling methods. *J. Database Manag.*, 15:73–86, 01 2004.
79. Allan Paivio. Dual coding theory: Retrospect and current status. 1991.
80. Certainly We Were Not Many and Yet We Were Sufficient. Estimating the number of subjects needed for a thinking aloud test. *Int. J. Human-Computer Studies*, 41:385–397, 1994.
81. Gay Costain. *Cognitive support during object-oriented software development: the case of UML diagrams.* PhD thesis, ResearchSpace@ Auckland, 2008.
82. Slava Kalyuga, Paul Ayres, Paul Chandler, and John Sweller. The expertise reversal effect. *Educational Psychologist*, 38(1):23–31, 2003.

CHAPTER 10

Article 4: Understanding Quality in Declarative Process Modeling Through the Mental Models of Experts

Understanding Quality in Declarative Process Modeling Through the Mental Models of Experts

Amine Abbad Andaloussi[1], Christopher J. Davis[2], Andrea Burattin[1], Hugo A. López[3,5], Tijs Slaats[3], and Barbara Weber[4] *

[1] Software and Process Engineering, Technical University of Denmark
2800 Kgs. Lyngby, Denmark – `amab@dtu.dk`
[2] University of South Florida, Saint Petersburg, FL, USA
[3] Department of Computer Science, University of Copenhagen, Denmark
[4] Institute of Computer Science, University of St. Gallen, Switzerland
[5] DCR Solutions A/S, Denmark

Abstract. Imperative process models have become immensely popular. However, their use is usually limited to rigid and repetitive processes. Considering the inherent flexibility in most processes in the real-world and the increased need for managing knowledge-intensive processes, the adoption of declarative languages becomes more pertinent than ever. While the quality of imperative models has been extensively investigated in the literature, little is known about the dimensions affecting the quality of declarative models. This work takes an advanced stride to investigate the quality of declarative models. Following the theory of Personal Construct Psychology (PCT), our research introduces a novel method within the Business Process Management (BPM) field to explore quality in the eyes of expert modelers. The findings of this work summarize the dimensions defining the quality of declarative models. The outcome shows the potential of PCT as a basis to discover quality dimensions and advances our understanding of quality in declarative process models.

Keywords: Process Model Understandability, Declarative Process Models, Model Quality, Personal Construct Psychology, Repertory Grid

1 Introduction

In the development of process-aware information systems (PAIS), process models are used for enactment and management purposes [4]. Besides their ability to provide a blueprint for process execution, process models are used for requirement elicitation, communication and process improvement. Process models are expressed using languages from either the *imperative* or *declarative* paradigm. While imperative models describe all the process executions explicitly,

---

* Work supported by the Innovation Fund Denmark project *EcoKnow* (7050-00034A), the Danish Council for Independent Research project *Hybrid Business Process Management Technologies* (DFF-6111-00337), and the European Union's Horizon 2020 research and innovation programme under the Marie Sklodowska-Curie grant agreement BehAPI No.778233.

declarative models rather specify the constraints guiding the overall process and allow any execution not violating the given constraints to occur. When dealing with rigid and repetitive processes, imperative languages are the best candidates. However, when it comes to knowledge-intensive processes where flexibly is an inherent requirement, imperative languages become unable to represent processes concisely. Alternatively, the constraint-based approach of declarative languages allows abstracting the details of specific process executions and modeling the general interplay of events. The flexibility of declarative languages comes at the cost of their understandability [16]. Considering the rich semantics of declarative languages and the different ways in which constraints can interact, it becomes hard for the reader to infer the process executions allowed by the model [34].

To support the understandability of declarative models, several hybrid representations extending models with textual annotations and simulations have emerged (review in [3]). Nevertheless, understandability challenges remained apparent [1]. Refining models to improve their quality is an alternative to overcome these limitations. While there is a rich body of literature investigating the quality of imperative models (e.g., [10, 17, 29, 30]), only a few contributions exploring the comprehension of declarative models exist (e.g., [20, 37]). A review by Corradini et al. [10] identified 50 guidelines addressing the quality of process models. However, many are limited to imperative languages and several of their focal constructs (e.g., gateways, pools and lanes, message events) are not relevant to declarative models. Similarly, the use of a single start event and the necessity to minimize concurrency in the model [10] are guidelines common to imperative modeling that counteract the constraint-based approach of declarative languages. Indeed, declarative models can have several entry-points [37]. Likewise, imposing a sequential-flow would need to overconstrain the declarative model, increasing its complexity - and reducing its understandability. In addition, modeling with constraints introduces conceptual challenges (e.g., hidden dependencies [37]), which are absent when modeling imperatively. Nonetheless, guidelines addressing the visual clarity of models (e.g., avoiding overlapping elements and line crossings) can be applied to both language paradigms. As a step towards the development of a more comprehensive framework for assessing the quality of declarative models, we use Personal Construct Theory (PCT) [24] to elicit quality dimensions used by experts when evaluating declarative models. Afterwards, we turn to the literature to discuss the similarities with existing guidelines and mark the key disparities requiring further investigation.

PCT directly fulfills our aim to elicit the criteria used by experts to judge model quality. It postulates that individuals develop a set of personal constructs (i.e., scales) to frame their experiences based on their similarities and differences [24]. In our context, the constructs offer scalar dimensions used by experts to differentiate the qualities of process models. Tapping into these constructs provide a means to articulate each expert's mental model, making the criteria by which model components are judged more tractable. Moreover, grounding our study in PCT overcomes many of the limitations of interpretive studies exploring the quality of process models, in particular those reliant on techniques such as interviews and think-aloud (e.g., [6, 37]). Insights obtained from interviews are usually bound by the interviewer's questions, leaving no chance to discover other relevant aspects beyond the repertoire of questions. As for think-aloud, it helps people to voice their thoughts out-loud and thus reveal their inner thoughts. However, as individuals tend to know more than they can readily articulate [12], part of their thought remains tacit and not readily evident in verbal utterances. PCT overcomes this limitation by removing the bounds of predetermination - the interview structure - offering in its place a framework for a series of comparisons. The similarities and differences between elements (e.g., those of process models) provide the basis for - and scope of - the technique. Through this comparison process, each individual's constructs can be articulated without constraint. Collectively, these benefits motivate our choice of PCT to articulate the constructs undergirding judgements of quality. Following analysis based on grounded theory [8], the constructs articulated are aggregated to propose a multi-dimensional framework for the assessment of declarative model quality.

Our contribution is twofold. Firstly, we develop a multi-dimensional framework that has the capacity to more comprehensively assess the quality of declarative models. Secondly, we demonstrate the potential of PCT in conducting interpretive analysis of process modeling. Our findings enhance the understanding of the dimensions of quality in declarative modeling and promote their use in industry. Moreover, these emergent dimensions of quality have the clear potential to support teaching of declarative modeling, helping students identify pertinent aspects requiring more attention when modeling processes declaratively. Finally, further adoption of PCT in the process modeling field would add to the stream of research exploring the mental models of practitioners. Sect. 2 presents the background, Sect. 3 introduces the related work, Sect. 4 explains the research method, Sect. 5 presents the findings, Sect. 6 discusses the findings and Sect. 7 wraps-up the key contributions and delineates the future work.

2 Background

DCR Graphs. DCR Graphs consist of nodes and edges: the nodes indicate *events*, the edges indicate *relations* between the events. Events can be assigned to *roles.* To maximize flexibility, events that are unconstrained can be executed at any time and any number of times. Events have a state marking, which is a tuple of three Boolean values: *executed*, *included* and *pending. Executed* indicates that the event has executed at least once in the past. *Included* indicates whether the event is currently relevant for the process: irrelevant (*excluded*) events cannot be executed, but also cannot constrain the execution of other events. *Pending* indicates that the event must be executed some time in the future, i.e. the event is a requirement that must be fulfilled before we can end the process. Pending events are generally referred as *required events.*

There are five basic relations. A *condition* restricts an event by stating that it cannot be executed before another event has fired at least once. *Milestones* constrain an event by stating that as long as a particular other event is pending, it cannot be executed. The *exclusion* and *inclusion* relations can be used to remove or add back an event from or to the process, effectively toggling event's included state. Finally, the *response* relation indicates that the execution of one event makes another event pending (i.e., required). The last three relations imply a dynamic behavior in the model as they are not constraints in the traditional sense, but rather capture *effects* that some events have on others. Relations and events can be combined together to model specific *behavioral patterns*.

Several extensions complement the core notation above. Hierarchy can be achieved through *nesting* [21], which allows one to group several events together (into a *nest event*), and then add a single relation to or from all of them. It simply acts as a shorthand for having a relation for each individual event and therefore does not add additional semantic meaning. The notion of *multi-instance sub-processes* [13] on the other hand, significantly extends the language by allowing one to model sub-process templates which can be instantiated many times. For example, a funding application round may consist of many individual applications, each application instance having their own unique internal state. Finally one can model the influence of contextual *data* on the process by adding *data expressions* to relations, indicating under what circumstances they should be activated [36]. For example, a response relation between "check expenses report" and "flag report" can be activated only if the amount exceeds a thousand euros.

Mental Models and Personal Construct Psychology. A mental model is an abstract representation of a situation or a system in the individual's mind [18]. Research on mental models addresses two aspects: their structure and change over time. Studies of the structure of mental models contribute to the theory of human reasoning and are used to evaluate individuals' decision making [23]. Change-oriented studies focus on dynamics where the system state changes over time. These studies investigate how individuals' mental models evolve and adapt [19]. In this work, we lean to the former, striving to articulate mental models whose structure reveals experts' judgement of declarative process models. The structure of the mental model - comprised of scalar constructs - provides direct insight into the criteria on which their assessment of quality is based.

To tap into individuals' mental models, we refer to the PCT theory of George Kelly [24]. Kelly assumed that individuals develop unique systems of interrelated personal constructs (i.e., scales), allowing them to understand and predict their surrounding world [24]. These personal constructs emerge from the individuals' past and ongoing experiences. Individuals organize and differentiate their experiences through judgement of similarities and differences, evolving a system of constructs, which they use to frame and predict the consequences of their own actions and interpret those of others [12]. The commonality of a system of constructs enables them to be used as a basis to explain interpersonal relations. This is particularly pertinent to personal experiences that share a cognitive medium or framework. PCT posits that individuals sharing common experiences can develop similar personal constructs [12].

In the view of Kelly, a personal construct is bipolar. It is composed of two ends (e.g., good versus bad). Eliciting constructs is challenging because individuals are generally unable to access the structure of their own cognitive system and verbalize their implicit knowledge [12]. *Repertory Grid* is a knowledge elicitation technique developed to help people identify and articulate their personal constructs [12, 24]. In a nutshell, the approach comprises a series of trials where a participant is asked to identify similarities and differences between different elements – such as process models in DCR Graphs. The result of each comparison is then used to articulate the participant's personal constructs and their meaning. A step-by-step explanation of the Repertory Grid process is provided in Sect. 4.2. Repertory Grid has been used in a wide range of domains (e.g., technology acceptance [12]). However, its potential has not yet been exploited in the field of process modeling. This work builds upon the PCT theory and adapts the Repertory Grid technique to derive a comprehensive framework delineating the dimensions used by experts to evaluate the quality of declarative process models.

Grounded Theory. Grounded Theory adopts a qualitative inductive approach to analyzing and conceptualizing data [8]. A multi-phase process of coding is a central to grounded theory, enabling the phenomena emerging from data to be identified and classified. Three coding techniques – *initial-coding*, *focused-coding* and *axial-coding* – are common [8]. Initial-coding highlights salient aspects in the data; focused coding allows these aspects to be grouped based on similarity of their traits, while axial-coding establishes relationships between the identified codes. Typically, a qualitative analysis starts with initial-coding, followed by focused-coding and finally axial-coding. In model comprehension studies, grounded theory has been used to analyze the verbal utterances of participants when interacting with different representations of process models (e.g., [1, 37]). Building on these works, our analysis uses the coding techniques of grounded theory to analyze the personal constructs verbalized by the experts throughout the different steps of the Repertory Grid.

3 Related Work

Model quality frameworks have emerged in different contexts. In conceptual modeling, guidelines addressing the use of graphical notations and the overall quality of conceptual models have emerged (e.g., [26,28,31]). In process modeling, a large body of literature focusing on the quality of imperative models exists (for an overview see the literature reviews in [10, 17]). In addition, a set of guidelines have been proposed on how to create process models of good quality (e.g., [27, 29, 30, 35]). However, when it comes to declarative languages, only a very limited number of studies exploring specific aspects of declarative models have emerged. Namely, the authors in [20] suggested that the comprehension of declarative models could be affected by the layout and the complexity of the used constraints. As for [37], the author suggested that modularization could support the comprehension of declarative models when solving a particular type of tasks.

Our study differs from the earlier works in several aspects. As opposed to [26, 28, 31] where guidelines are generic to any model-based representation, our work emphasizes declarative models, in particular those in DCR graphs, providing a closer examination of the quality dimensions relevant for that matter. With regards to [10, 17, 27, 29, 30, 35], many of the proposed guidelines either do not apply to declarative models or need further investigation to ensure their applicability (cf. Sect. 1). Alternatively, our research bases its analysis on declarative models and compares to related work to highlight the similarities and disparities between imperative and declarative guidelines (cf. Sect. 6). When it comes to studies looking into declarative process models, we argue that model quality was not well emphasized. Instead, the focus was on exploring the use of declarative models [20] or assessing the impact of modularization [37] on the performance of users. Conversely, our work emphasizes the quality of declarative process models and aims at providing a multi-dimensional quality framework to further promote their use in practice. Besides, our study design (based on PCT, cf. Sect. 4) differs from the existing qualitative designs as explained in Sect. 1.

4 Research Method

This section introduces our research method including the research question (cf. Sect 4.1), data collection (cf. Sect. 4.2) and analysis procedures (cf. Sect. 4.3).

4.1 Research Question

This work addresses the need for a comprehensive framework allowing to evaluate the quality of declarative process models, particularly DCR Graphs. Our research question is formulated as follows: **Which quality dimensions are used by experts when comparing DCR Graphs?**

4.2 Data Collection

Data was collected using a step-wise approach underpinned by PCT. The following sections explain our data collection process in detail, introduce the research setting, and describe the materials used in the study.

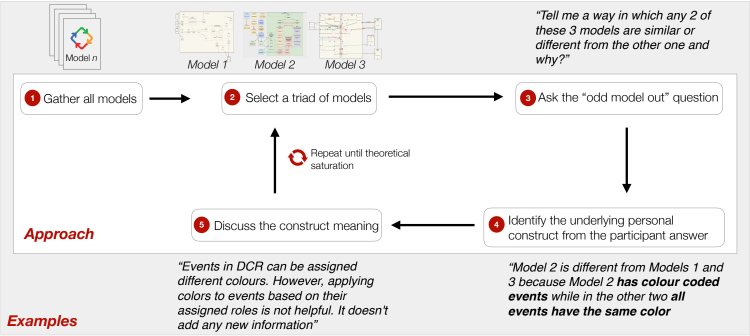
Approach. Following the theoretic position set out in Sect. 2, we use the Repertory Grid to identify the constructs used by experts to evaluate the quality of DCR Graphs. The elicitation process is initiated by the *selection of a set of elements* referring to different instances of a universe of discourse [24]. Repertory Grid studies use different types of elements. In clinical contexts, elements are usually represented as roles (i.e., people); however, in other studies, elements are represented as working tasks [12]. In our study, we consider the elements as models provided by modelers with different levels of expertise. *Collecting the models* representing the elements of the grid is, then, the first phase of our data collection. To this end, we have shared a process description with a set of participants and asked them to design the corresponding model in DCR Graphs. The resulting models are available in our online repository [2].

Once the models defining the elements of the grid have been collected, we move to the second phase of our data collection, where participants recruited for their expertise evaluate the quality of the collected models. This phase begins by *eliciting of personal constructs*. Through a series of trials, the participant is given a triad (i.e., set of three) of models and asked, following the minimum context form described by Kelly [12, 24]), to *(1)* identify the "*odd model out*" (i.e., the model that differs from the other two models of the triad) *(2)* and explain "*why*", that is to say, what –in her terms – makes it odd. This articulates one dimension of the scale used to differentiate the models (elements). The participant is then asked what –if anything– makes the remaining (non-odd) elements similar. Often, this is a simple negation: for instance, a triad composed of 3 process models might be differentiated because one model *has color coded events*, while in the other two *all events have the same color*. In this sense, the construct defined with the poles *has color coded events* versus *all events have the same color* is an example of a participant's personal construct. A construct is thus articulated as two distinct poles drawn from the difference between the odd model and the similarity of the other two models.

The identification of personal constructs is usually complemented by a discussion of the meaning of the constructs to the participant. The discussion is moderated using *laddering up* and *laddering down* techniques used respectively to elaborate or abstract the insights offered by the participant, further articulating their relevance [12]. The same triad approach is repeated until a theoretical saturation of constructs is reached. Rather than data saturation, where all possible triads should be visited, we follow a theoretical saturation approach, striving to provide the participant with new triads until no more new constructs emerge. On average, most constructs were articulated after 7 triads, which falls within the same range of triads generally used to identify the most salient constructs [11]. Fig. 1a summarizes the process of eliciting personal constructs.

Following the identification of constructs, the participant is given a grid where columns represent the collected models and rows show the identified constructs. During this process, the participant is allowed to review and edit her constructs before being asked to *rate each of the models based on the identified constructs*. The literature discusses different rating methods [12], in our study, we use a five-point scale following the insights in [12]. As the constructs usually emerge from comparisons within triads of models, some constructs might not apply to all models. In such a case the participant is told to skip these particular grid cells. Analysis of the numeric ratings enables the grid to illustrate underlying but unseen associations between elements and constructs and thus their meaning using concrete terminology drawn from the participants 'world', which in turn supports the analysis of these personal constructs. A fragment of a Grid is illustrated in Fig. 1b. The collected grids are available in our online repository [2].

The *talkback interview* is the last step. It aims at reflecting the overall process and scrutinize the personal constructs based on the obtained qualitative insights and the grid ratings. While some studies conduct further statistical analyses to investigate the correlations between constructs and elements, our



(a) The process of eliciting the participants' personal constructs

Personal Construct Poles	**Elements**						**Personal Construct Poles**
	Model 1	*Model 2*	*Model 3*	*Model 4*	*Model 5*	*Model 6*	
Has color coded events	5	4	2	5	3	1	*All Events have the same color*
...	3		2		5	1	....

(b) Fragment of a Repertory Grid

Fig. 1: Illustrations of the different steps of the Repertory Grid approach

work rather focuses on the insights obtained throughout the different steps of the Repertory Grid and analyzes them following grounded theory. To keep track of these insights, the conversations with the participants were fully recorded.

Participants. To collect the models representing the elements of the grid, we have recruited 13 participants with different levels of expertise in DCR Graphs. Novice participants (3 students) have taken a BPM course where they have been introduced to process modeling in general. Intermediate participants (4 students) have been familiarized with DCR Graphs for at least one semester. Whereas expert participants (2 professors, 2 postdocs and 2 industry practitioners) are more deeply immersed through their use of and research into DCR Graphs. The heterogeneity of participants enabled us to explore the range of model complexities and reflect different modeling practices employed by users with different levels of expertise. This heterogeneity also provides the basis to allow differences between novices and experts - and novice models and expert models - to emerge.

To evaluate the models, we used 4 experts among the pool of participants in the first phase. Each expert was exposed to the 12 models collected from the other participants and her model as well. Including experts' own models in the comparison gave them the opportunity to reflect on their models (compared to others) which in turn enriched the analysis. Overall, 94 bi-polar personal constructs were elicited from the models [2].

Material. The process description used to collect the models representing the elements of the grid is inspired by a real-world use-case study presented in [15]. The process description (cf. online repository [2]) was shared with 13 participants, who were asked to design the corresponding process model in DCR Graphs.

4.3 Data Analysis

The analysis started by listening to the audio recordings of the repertory grid procedure, timestamping the periods where each of the constructs was discussed, and then taking notes of the collected insights. Here, the verbal utterances provided by each participant were related directly to the ratings of the relevant model in the repertory grid providing concrete, context-specific articulations of the participant's insights. Afterwards, we turned to grounded theory to investigate the participants' constructs and their meanings. To reduce subjectivity during the coding process, we recruited two coders. We followed the code-confirming strategy [25] to distribute the tasks between the primary and the secondary coders. The primary coder was responsible for conducting the first round of coding, while the secondary coder was recruited to critically scrutinize the codes and trigger discussions to improve the coding. Both coders are researchers within the BPM field. For each grid, the primary coder conducted the first round of initial-coding to the participant's constructs [7] based on the constructs' poles. In case the poles were not clear, the primary coder referred to the collected notes. Afterwards, the secondary coder reviewed the initial-coding and performed the second round of coding, which was, in turn, discussed by both coders to reach an agreement. Next, the constructs obtained from all the participants were combined and subjected to focused-coding [7] grouping repeating and overlapping initial codes to identify the commonality or focus among the concepts articulated. The resulting codes reveal the different *dimensions* used by the participants to evaluate the quality of declarative process models. The relationships between the revealed dimensions were elaborated using axial coding [7]. Here, the revealed dimensions were organized according to recurrent themes and then categorized. This phase was conducted in 2 rounds by both coders, followed by a discussion where the final codes were agreed. An excel sheet illustrating this process is available as part of our online repository [2]. The resulting categories, themes and dimensions are presented in Sect. 5.

5 Findings

The analysis of the constructs allowed the identification of seven themes organized into 2 categories. Sections 5.1 and 5.2 present the themes associated with the semantic qualities and pragmatic qualities of process models respectively.

5.1 Semantic Qualities

Semantics denotes the ability of the model to make true statements about the way the business process operates in the real-world [35]. The semantics of a model is a relative indicator of quality as the model behavior is subjective to the process specifications. The analysis of experts' personal constructs, drawn from their interpretation of the models, gave rise to 4 themes overarching a number of dimensions capable of assessing the semantic quality of DCR Graphs: these are *modeling behavior, modeling patterns, modeling events* and *modeling data.*

Modeling Behavior. Within this theme, several dimensions have emerged. *Comprehensiveness of behavior* is identified throughout our analysis of personal constructs. Here, experts used this dimension to evaluate the completeness of the model. When it comes to the alignment between the process specifications and the model behavior, the experts elicited the *presence of behavioral errors* dimension to assess the validity of the behavior supported by the model.

Flow-based versus *declarative modeling* is a relevant dimension used by the experts to evaluate flexibility. They identified a spectrum of modeling behaviors ranging from very flexible to over-restricted ones. Overall, the experts asserted that declarative models should support parallel behavior and avoid being restrictive. Nevertheless, they also advised avoiding both extremes (being too flexible or too restrictive) and advised to rather comply to the *process specifications.*

Modeling of required events is another relevant dimension identified by the experts. This dimension evaluates the modeling of events that must eventually be executed in the process. These events are regarded as goals which must be fulfilled in any execution [22]. Identifying these events in the specifications and modeling them correctly are important criteria to model behavior consistently. In DCR Graphs, required events can be modelled by assigning a specific marking to events at design time or by using the response relation (cf. Sect. 2).

The experts identified the dimension *Modeling of end-events* to assess whether the model allows termination. In DCR Graphs, end-events refer to events whose execution disable the rest of events in the model from executing. While some experts recommended to model termination, other experts asserted that one cannot generalize that all processes should incorporate termination. In some cases, process specifications require processes to be *suspended* rather than terminated, leaving the possibility to resume them at any point in time. For such processes, only the no-longer relevant events should be removed from the process before suspension. Similarly, the experts identified the dimension *Modeling of start-events* (i.e., events initiating the process) to assess whether the models identify the process start-events appropriately. In this respect, some experts advised using a unique start-event, while other experts affirmed that this depends on the process specifications. Nonetheless, experts advised checking whether the non-constrained events in the model are good candidates for being start-events to the process, if not, then these events must be constrained by others to prevent their occurrence when the process is first initiated.

Additionally, the *Multi-instance processing* dimension emerged to compare the extent to which multi-instance sub-processes are supported (cf. Sect. 2). From this perspective, the experts noticed that most of the models do not comply with the given process specifications as they do not offer the possibility to indicate the parts which can be executed multiple times concurrently.

The *Modeling against IT silliness* dimension addresses the experts' felt need to assess the flexibility of the models in tackling failures that prevent occurred events from being registered by the PAIS. In this context, the distinction between *unlawful behavior* (i.e., the behavior violating the constraints of the process) and *impossible behavior* (i.e., a behavior which would never occur in the real-world) has emerged. While the former is crucial to avoid, the latter can be tolerated assuming that the PAIS might fail to register some non-value adding events at their occurrence (e.g., granting a loan without signing the contract must never be allowed, whereas, signing the contract without receiving it, could be tolerated by the model assuming that the PAIS failed to register that event).

The *purpose of the model* is a dimension used by the experts to evaluate the granularity of the models. Accordingly, the level of detail exposed by the scope or bounds of the business process can be adjusted to fit the intended purpose (e.g., enactment, management). The identification of the model purpose is a crucial aspect because it goes beyond the semantic qualities of the model also to affect the pragmatics of the model. In that sense, a model intended for enactment can be hard to interpret if used for management purposes.

Modeling Patterns. Modeling patterns denote the set of mechanisms used to represent specific behaviors when modeling processes. The elicited insights focused on the *use of standard patterns*, which encompass the conventional modeling patterns advised for modeling different behaviors. For experts, standard patterns provide a clear representation of the intended model behavior. The use of standard patterns also reoccurred while inspecting the way modelers represented common behavior, exceptional behavior, and termination.

The dimension *Condition-response* versus *Include-exclude patterns* emerged when comparing the common behavior represented in the models. The *condition* and *response* relations can be used together to model a wide range of specifications. However, a similar behavior can be achieved using *exclude* and *include* relations, which was recurrent in many models. During the discussion, the experts advised adhering to the *condition-response* pattern when modeling common behavior for the following reasons: *(1)* The dynamic behavior of the *include* and *exclude* relations (cf. Sect. 2) is more likely to create hidden dependencies between events, adding unnecessary complexity to the model. *(2)* The *include* and *exclude* relations are rather used for modeling exceptions and termination.

The dimension *Treatment of exception pattern* assesses whether the modeler uses the appropriate pattern to treat exceptions clearly. For the experts, exceptional events are not part of the main process and thus they should initially be excluded in the model and included (using the include relation) only when exceptions occur. Likewise, the dimension *Use of termination pattern* addresses whether termination is modeled using the appropriate pattern. Here, the experts recommended grouping events into a nest event (cf. Sect. 2) and add one exclude relation from the end-event to the nest event.

Modeling Events. The experts used the *Role assignment* dimension to check the assignment of roles to events and asserted that it is crucial for clarifying “who is doing what?”, which in turn supports better traceability and access control.

Use of intermediate events is a pertinent dimension. In DCR, intermediate events denote the events used to enforce specific behaviors, without being explicitly mentioned in the process specifications. Intermediate events can be used to automate some actions or to model decisions. For the experts, although their use might be necessary (e.g., for implementation), intermediate events can hinder the understandability of the model and should be avoided whenever possible.

Besides, the *implicitness of events* dimension was introduced to evaluate whether all the events mentioned in the process specifications are explicitly represented in the model. Indeed, some modelers merged several events into one. For the experts, modelers should ensure a one-to-one correspondence between the events of the process specifications and those represented in the model.

Modeling Data. The dimension *Encoding decisions explicitly or using data expressions* was used by the experts to evaluate whether decisions are encoded using intermediate events or using data expressions. As mentioned in Sect. 2, data events allow assigning values to variables, which in turn are used in the evaluation of data expressions. Following the experts, the activation of the DCR relations in a model can be controlled by assigning them data expressions. At run-time, if the expression evaluates to *true*, then the semantics of the relation applies in the model, otherwise, it doesn't. Data expressions can be difficult to interpret. However if used purposefully for modeling decisions, they can reduce the complexity of the model (e.g., by removing intermediate decision events).

Besides, the experts identified the dimension *Appropriate choice of data types for data variables* to indicate cases where the data types of variables were not correctly chosen. Here, the experts highlighted the necessity of choosing a data type which infers meaning about the use of the variable it represents.

The *Local/global effect of data variables* dimension emerged to describe whether a data variable is evaluated immediately after being assigned a value (using a data event), or postponed to a later stage of processing. On that matter, the experts recommended evaluating data variables immediately after assigning them values, making the correspondence between the data event and its subsequent evaluation clearer. However, depending on the process specifications, an immediate evaluation of data variables is not always feasible. In this case, the experts advised a consistent naming of data events and data variables, making the correspondence between both easily perceived (cf. Sect. 5.2).

5.2 Pragmatic Qualities

Pragmatics denotes the correspondence between the model and the reader's understanding of it [35]. The pragmatic qualities of a model do not formally affect its behavior. However, they might have direct consequences on the use of the model as a communication artifact. The experts' meanings revealed 3 themes related to pragmatic qualities: *Model Layout, Event Layout, Data Layout.*

Model Layout. The experts used the dimension *Alignment and positioning of elements* to appraise the way models are laid out. They highlighted the extraneous visual complexity raising from models where elements (i.e., events, relations) overlap, and advised a careful alignment and spacing of events. Here, two strategies were used: the former evaluates whether the events assigned to the same role are aligned along the same vertical axis, while the second strategy assesses whether the events are aligned following their likely order of occurrence during execution. For the experts, these strategies could improve the pragmatic quality of the model. In addition, the experts looked into the way models were oriented and suggested a left-to-right or top-to-bottom orientation, indicating that start-events should be positioned at the left-most top-most part of the model.

The *grouping of events* dimension evaluates the way events are grouped in the model. Nest events (cf. Sect. 2) allow gathering events belonging to the same phase or assigned to the same role. With a preference for phase-based nesting, the experts associated the use of nesting with an enhanced understandability of the model. In the same vein, multi-level hierarchy was raised by experts to emphasize the benefits of going beyond a single level of nesting.

Visual conciseness focuses on the overall clarity of the model. This dimension was defined by the previously mentioned aspects e.g., alignment and grouping of events, but also in relation to the optimized use of constraints and the absence of intermediate events. These characteristics embrace both pragmatic and semantic qualities, showing that the themes and dimensions emerging within both categories influence the experts' perception of visual conciseness.

Event layout. The experts emphasized particularly the internal pragmatics of events. The dimension *Meaningful naming of events* was used to assess the meaningfulness of events' names. For experts, events should be assigned comprehensible names which can be easily traced back to the process specifications.

Furthermore, the experts used the dimension *Verb-object* versus *noun-based naming of events* to evaluate the phrasing of the events' names. Here, they recommended a verb-object phrasing, except for the intermediate events used for modeling decisions, where a noun-based format could be acceptable.

Color coding was another identified dimension. Although, DCR allows assigning colors to events, some experts were confused by the meaning of these colors, and asserted that they are hard to interpret when no explicit legend is provided. Hence, several experts suggested avoiding to color events.

Data Layout. The dimension *Correspondence between variable names and data events' names* was used by the experts to evaluate whether the data event altering the value of a data variable can be easily recognized in the model. For experts, data events and data variables should be assigned the same name because data variables might not be evaluated immediately after being assigned a value. Hence, with the lack of a clear matching between a variable name and its corresponding data event's name, it becomes hard for the reader to infer the variable's value when being evaluated in a data expression as all the previously executed data events could presumably change the value of that data variable.

6 Discussion

The dimensions identified by the experts share many similarities with the existing imperative process modeling guidelines. For instance, comprehensiveness of behavior and presence of behavioral errors (two of the identified semantic qualities) relate to the notions of completeness (i.e., the coverage of the relevant statements of a particular domain) and validity (i.e., the correctness of the statements in the model) discussed in [28]. Moreover, the importance of designing models fitting their intended purpose (i.e., enactment, management) both in terms of granularity and target audience was not only recognized by our experts, but also emphasized in [28]. In terms of pragmatic qualities, the insights about the alignment and positioning of elements intersect with the findings in [10, 30], while the recommendations about assigning meaningful names to events and phrasing them following a verb-object format have been discussed in [10, 30]. Regarding the use of colors to mark events, there was no agreement between experts. This concurs with literature on the usage of color in the context of imperative processes which is also inconclusive [5, 10].

The use of standard patterns is among the pertinent dimensions, which experts argued it enhances the understandability of the model. While catalogues of patterns showing how to model certain re-occurring problems exist for imperative models [14], we cannot currently rely on such resources when modeling declaratively. Additional research is needed to elicit a catalogue for DCR Graphs and to empirically evaluate its impact on model quality. Our findings show that the general idea of using decomposition to reduce process model complexity is shared with imperative models [37]. However, additional guidelines – on when and how to decompose declarative models – are missing. Decomposition in imperative models involves identifying particular points in the flow where a complex behaviour can be abstracted into an individual step with a single entry and exit point. This is not as easy in declarative modeling, where different parts of the model may interact in different ways, making it challenging to find clear distinctions between the entangling constraints of the model. There is also a need for empirical research on the impact of modularization on the quality of declarative models. Existing research [37] suggests that modularization enables abstraction and information hiding, which in turn supports the comprehension of the model. Contrarily, modularization also risks fragmentation, giving rise to split-attention effects and a need for integration between different parts of the model.

Existing guidelines on the usage of gateways for modeling decisions are not applicable to declarative models, including DCR graphs. Experts mentioned the modeling of decisions using either intermediate events or data expressions. The use of events to model decisions would lead to construct overload as a single notational element is being used to represent multiple concepts (i.e., actions and decisions). Existing research states that construct overload impacts the understandability of the model negatively [31]. Alternatively, experts suggested modeling decisions using data expressions. However, the implications of using data expressions on the understandability of declarative models are questionable and require additional research. Regarding the modeling of start-events, existing guidelines [30] advise use of a single start-event. While some experts agreed, others questioned the general applicability of this guideline and suggested that it depends on the process. Due to the constraint-based approach of declarative languages, any non-constrained event is a possible entry-point to the process. This makes modeling of start-events in declarative languages more complex than imperative languages since in declarative modelers one must check all non-constrained events to ensure that they are good candidate start-events for the process or constrain them to prevent their occurrence when the process is first initiated.

While several insights agree with the literature on imperative process models, our study identified some contradictions. For instance, our findings promote the concurrency of behavior in declarative models, whereas existing guidelines [10] advise minimizing concurrency when modeling imperatively. Moreover, existing guidelines [32] assume that processes should eventually terminate. Conversely, our insights relax this assumption by evoking the possibility of suspension instead of termination. However, little is known about when to use what, which necessitates detailed guidelines. Moreover, while the use of single end-events is recommended to ensure understandable models [30], the impact of modeling processes without explicit end-events is yet to be explored.

The results of this study have impacts on research, education and practice. The insights obtained advance our understanding of quality in declarative models. While several of the findings concur with prior research on imperative modeling, our study also revealed several dimensions where further investigation is required. The positive effects of standard patterns on both quality and comprehension of declarative models suggest a potential hypothesis worthy of test in the light of the existing theory. A further hypothesis might address effects of applying modularization on the understandability of declarative models. Moreover, the applicability of PCT in process modeling paves the path for new studies exploring the mental models of practitioners when dealing with different aspects of process models. With regards to education, our findings support the teaching of declarative process modeling (particularly in DCR Graphs) by providing a set of dimensions allowing students to focus their attention on the pertinent quality aspects to improve their design of declarative models. Our findings also have implications for practice. Several of the identified semantic qualities (relating to modeling of events and data) and pragmatic qualities (related to model, event and data layouts) can be automatically inferred from the model and thus could be implemented by tool vendors to assess the quality of process models at design time offering the potential of customized tool-support for modelers.

Limitations. Our research has some limitations. Our sample is relatively small: however, in common with other Repertory Grid studies (e.g., [9,33]) the scale and richness of the elicitation process gave rise to over 400 numeric data points, highlighting both the cognitive focus and demand of the approach, which required some 4-5 hours per session. Another limitation might arise through bias during the coding procedure. To minimize this risk, we recruited a secondary coder who was purposefully critical of the coding of the primary coder. Finally, our results do not address syntactical qualities since the models were all designed using a tool (i.e., `dcrgraphs.net`) which automatically resolves syntax-dependent errors.

7 Conclusion and Future Work

This work investigates the quality of declarative process models. The results present a set of quality dimensions identified by experts in DCR Graphs. Similarities with existing guidelines highlight qualities shared with imperative models – while clear differences identify candidate aspects worthy of further investigation. Future work could subject the different qualities to further theoretical and empirical investigation. Several hypotheses have already emerged, as noted above. Moreover, our data could be used to investigate how different quality dimensions affect each other. The models provided by the different groups of participants could be further analyzed to discern patterns characterizing the modeling of novices, intermediates and experts, which in turn could guide the profiling of modelers at run-time and optimizing tool support. Our approach also offers sound potential to contribute to studies that explore the mental models of practitioners and their interaction with process models.

References

1. Abbad Andaloussi, A., Burattin, A., Slaats, T., Petersen, C.A., Hildebrandt, T.T., Weber, B.: Exploring the Understandability of a Hybrid Process Design Artifact Based on DCR Graphs. In: BPMDS (2019)
2. Abbad Andaloussi, A., Davis, C.J., Burattin, A., López, H.A., Slaats, T., Weber, B.: Online repository: used material, collected data and full analysis (March 2020), `http://doi.org/10.5281/zenodo.3724609`
3. Andaloussi, A.A., Burattin, A., Slaats, T., Kindler, E., Weber, B.: On the declarative paradigm in hybrid business process representations: A conceptual framework and a systematic literature study. Information Systems 91, 101505 (2020)
4. Becker, J., Rosemann, M., von Uthmann, C.: Guidelines of Business Process Modeling pp. 30–49 (2000)
5. Bera, P., Soffer, P., Parsons, J.: Using eye tracking to expose cognitive processes in understanding conceptual models. MIS Quarterly 43(4), 1105–1126 (2019)
6. Bernstein, V., Soffer, P.: How does it look? exploring meaningful layout features of process models. In: International Conference on Advanced Information Systems Engineering. pp. 81–86. Springer (2015)
7. Bryant, A., Charmaz, K.: The SAGE Handbook of Current Developments in Grounded Theory. SAGE Publications (2019)
8. Charmaz, K.: Constructing Grounded Theory. Introducing Qualitative Methods series, SAGE Publications (2014)
9. Christie, D.F., Menmuir, J.G.: The repertory grid as a tool for reflection in the professional development of practitioners in early education. Teacher Development 1(2), 205–218 (1997)
10. Corradini, F., Ferrari, A., Fornari, F., Gnesi, S., Polini, A., Re, B., Spagnolo, G.O.: A guidelines framework for understandable bpmn models. Data & Know Engineering 113, 129–154 (2018)
11. Curtis, A.M., Wells, T.M., Higbee, T., Lowry, P.B.: An overview and tutorial of the repertory grid technique in information systems research. Communications of the Association for Information Systems 23(3), 37–62 (2008)
12. Davis, C.J., Hufnagel, E.M.: Through the eyes of experts: A socio-cognitive perspective on the automation of fingerprint work. MIS Quarterly 31(4) (2007)

13. Debois, S., Hildebrandt, T.T., Slaats, T.: Replication, refinement & reachability: complexity in dynamic condition-response graphs. Acta Informatica 55(6) (2018)
14. van Der Aalst, W.M., Ter Hofstede, A.H., Kiepuszewski, B., Barros, A.P.: Workflow patterns. Distributed and parallel databases 14(1), 5–51 (2003)
15. Elgammal, A., Turetken, O., van den Heuvel, W.J., Papazoglou, M.: Formalizing and appling compliance patterns for business process compliance. Software & Systems Modeling 15(1), 119–146 (2016)
16. 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: Proceedings of EMMSAD. pp. 353–366 (2009)
17. Figl, K.: Comprehension of procedural visual business process models. Business & Information Systems Engineering 59(1) (2017)
18. Forrester, J.W.: Industrial dynamics. 1961. Pegasus Communications (1961)
19. Groesser, S.N., Schaffernicht, M.: Mental models of dynamic systems: taking stock and looking ahead. System dynamics review 28(1), 46–68 (2012)
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 and Systems Modeling 15(2), 325–352 (2016)
21. Hildebrandt, T., Mukkamala, R.R., Slaats, T.: Nested dynamic condition response graphs. In: Proceedings of Fundamentals of Software Engineering (2011)
22. Hildebrandt, T.T., Mukkamala, R.R.: Declarative Event-Based Workflow as Distributed Dynamic Condition Response Graphs. EPTCS 69, 59–73 (2011)
23. Johnson-Laird, P.N.: Mental models and deduction. Trends in cognitive sciences 5(10), 434–442 (2001)
24. Kelly, G.A.: The psychology of personal constructs norton. New York (1955)
25. King, N., Horrocks, C., Brooks, J.: Interviews in Qualitative Research. SAGE Publications (2018)
26. Krogstie, J.: Model-based development and evolution of information systems: A Quality Approach. Springer Science & Business Media (2012)
27. Krogstie, J.: Quality in Business Process Modeling. Springer, Cham (2016)
28. Lindland, O.I., Sindre, G., Solvberg, A.: Understanding quality in conceptual modeling. IEEE software 11(2), 42–49 (1994)
29. Mendling, J.: Metrics for process models: empirical foundations of verification, error prediction, and guidelines for correctness. Springer (2008)
30. Mendling, J., Reijers, H.A., van der Aalst, W.M.: Seven process modeling guidelines (7pmg). Information and Software Technology 52(2), 127–136 (2010)
31. Moody, D.: The "physics" of notations: toward a scientific basis for constructing visual notations in software engineering. IEEE Transactions on SE (2009)
32. Moreno-Montes de Oca, I., Snoeck, M.: Pragmatic guidelines for business process modeling. Available at SSRN 2592983 (2014)
33. P Gary John, K.M.: Developing an expert support system for tender enquiry evaluation: A case study. European Journal of Operational Research 56 (1992)
34. Pesic, M.: Constraint-based workflow management systems : shifting control to users. Ph.D. thesis, TU Eindhoven (2008)
35. Reijers, H.A., Mendling, J., Recker, J.: Business process quality management. In: Handbook on Business Process Management 1, pp. 167–185. Springer (2010)
36. Slaats, T., Mukkamala, R.R., Hildebrandt, T., Marquard, M.: Exformatics declarative case management workflows as dcr graphs. In: BPM proceedings (2013)
37. Zugal, S.: Applying Cognitive Psychology for Improving the Creation, Understanding and Maintenance of Business Process Models. Ph.D. thesis, University of Innsbruck (2013)

Chapter 11

Article 5: Assessing the Complexity of Declarative Process Models Using Model-based Metrics

Assessing the Complexity of Declarative Process Models Using Model-based Metrics

Amine Abbad Andaloussi[1], Andrea Burattin[1], Tijs Slaats[2], Ekkart Kindler[1],
and Barbara Weber[3] ⋆

[1] Software and Process Engineering, Technical University of Denmark
2800 Kgs. Lyngby, Denmark
[2] Department of Computer Science, University of Copenhagen, Denmark
[3] Institute of Computer Science, University of St. Gallen, Switzerland

Abstract. The understandability of process models is a quality and a key requirement to foster the communication between process stakeholders and ensure a good understanding of business processes within organizations. While a plethora of metrics have been proposed to measure the complexity of imperative process models and assess their understandability, less research has been devoted to the declarative ones. This paper addresses this shortcoming by proposing and empirically evaluating a set of metrics for declarative process models. The outcome of this paper delivers quantifiable means to evaluate the complexity of declarative models and appraise their understandability from a human-cognitive perspective. The proposed metrics can be easily implemented and embedded in existing declarative process modeling tools to provide feedback about the complexity of process models at design time.

Keywords: Complexity metrics · Process model understandability · Declarative process model · Cognitive load

1 Introduction

Process models provide a key instrument for the design and development of today's Process-Aware Information Systems (PAIS). Their use is vital for the enactment and management purposes [6]. On an enactment level, process models provide a blue-print for process execution [17], while on a management level, process models can be used to elicit the requirements underlying business processes and illustrate the way they operate in the real-world [17, 56]. Moreover, process models can support the communication and collaboration between domain experts and IT specialists [16], and can be used for benchmarking, process improvement and optimization [17].

The use of process models for both enactment and management requires formal languages that are not only interpretable by machine but also understandable by humans. Several process modeling languages have emerged to attain this aim. These languages can be organized into the imperative-declarative paradigm

---

⋆ Work supported by the Innovation Fund Denmark project *EcoKnow* (7050-00034A),

spectrum [19]. While process models created using imperative languages explicitly depict the execution paths allowed in the model, their declarative counterpart abstract the execution paths from the model and rather emphasize the constraints governing the interplay between the process activities [19, 58]. The flow-based nature of imperative languages makes them adequate for representing predefined and repetitive processes (e.g., check-in procedure in airports), whereas the constraint-based approach adopted by declarative languages provides better means to represent flexible processes (e.g., health-care processes) concisely.

The quality of process models has been investigated with regards to both imperative [6,31,32,41] and declarative languages [2]. Therein, a series of guidelines and modeling practices have been suggested to support modelers during process modeling and help them to design understandable models with enhanced quality. However, many of the existing quality frameworks lack quantifiable means for assessing the understandability of a process model. In imperative process modeling, a wide range of complexity metrics has been proposed to address this shortcoming (literature reviews in [36, 57]). However, when it comes to declarative modeling, to the best of our knowledge, only a single study [35] has looked into that aspect.

Providing complexity metrics for declarative process models is needed to evaluate their design and make accurate estimations about their understandability. These estimations can help to compare different process diagrams and infer when structural changes are required to improve the quality of existing models.

In this paper, we propose two contributions. Firstly, we provide a set of complexity metrics for declarative process models. Secondly, we report the results of an empirical study testing the impacts of the factors captured by these metrics on the understandability of declarative models. Regarding the first contribution, we explore the existing literature about complexity metrics for imperative process models. Therein, we discern (a) metrics addressing the complexity of the sequence-flow encoded in the model (e.g., concurrency metrics [40]), and (b) metrics addressing the graph structure of the model (e.g., density metrics [40]). We conjecture that, due to the constraint-based approach of declarative languages, sequence-flow metrics cannot apply to declarative models, whereas structural metrics may apply as they capture generic features that can be relevant to other graph-based representations. Following this distinction we select a set of candidate metrics, which we further investigate to lay out the theoretical foundations allowing to derive new metric variants that can capture the complexity of declarative process models. We focus on a particular declarative language i.e., Dynamic Condition Response (DCR) [27]. Our choice of DCR is motivated by the availability of tools at the industrial level [37] and wide range of documented applications in the real-world [13, 14, 26, 67]. Subsequently, we define four complexity metrics (i.e., *size*, *density*, *separability* and *relation variability*), that we validate empirically in the following contribution.

In the second contribution, we formulate a set of hypotheses about the impacts of the factors captured by the proposed metrics on model understandability, which we address from a human-cognitive perspective. Herein, we turn to the field of cognitive psychology. In particular, we focus on the concept of *cognitive load*, that is defined as the amount of load on imposed on the human working memory [50] (cf. Section 2.1). We use three indicators of cognitive load (i.e.,*perceived difficulty*, *comprehension accuracy* and *response time* [8]), which we investigate to assess the difficulty experienced by people when engaging with declarative process models varying in terms of their complexity. The results of the empirical study demonstrate the impacts of the covered factors on cognitive load and motivate the adoption of the proposed metrics in practice.

The paper is structured as follows: Section 2 presents the background and related work. Section 3 introduces the proposed metrics. Section 4 explains the design of the empirical study. Section 5 reports the findings of the study. Section 6 discusses the main findings. Section 7 presents the limitations of study. Section 8 concludes the paper.

2 Background and Related Work

This section provides the background on the concept of cognitive load (cf. Section 2.1) and the DCR language (cf. Section 2.2). Moreover, the section presents the related work about complexity metrics (cf. Section 2.3).

2.1 Cognitive Load

Humans' memory has been subject to extensive research in the field of cognitive psychology [69]. In particularly, two types of memory have been studies: the (1) *long term memory* representing a store capable of holding information like events, experiences, concepts and rules for long time periods or indefinitely [5,69], and the (2) *working memory* denoting a store of limited capacity capable of holding and processing information for a short time span [5, 69].

The cognitive load theory (CLT) addresses the properties of the working memory. The theory defines the concept of cognitive load as *"a multi-dimensional construct representing the load imposed on the working memory during [the] performance of a cognitive task"* [8, 50, 51]. The theory also posits that humans' working memory has a limited capacity [50]. As a result, the amount of information which an individual can process at a time is bound to a certain limit (i.e., usually 7±2 items at a time [42]). Following CLT, working memory may become a bottleneck when dealing with difficult tasks. Therein, humans' cognitive load can reach high levels, which in turn can affect their performance negatively and drive them to make wrong decisions [8, 18, 34, 47].

Several indicators of cognitive load have been proposed in the literature [8]. These indicators can be organized into subjective (e.g., perceived difficulty), performance (e.g., comprehension accuracy, response time), behavioral (e.g., gaze patterns, user interactions) and physiological (e.g., galvanic skin response, heart rate).

As mentioned in Section 1, this paper comprises an empirical study investigating the impacts of several (model-based) factors on the understandability of declarative process models, which we address from a human-cognitive perspective. To this end, we rely on the aforementioned subjective and performance indicators to estimate people's cognitive load when conducting comprehension tasks on process models with different levels of complexity.

2.2 DCR Language

DCR is a declarative process modeling language [27] developed in collaboration between research groups in academia and industrial partners in Denmark. The language is supported by a process modeling platform (i.e., DCR Portal[4]) and commercial PAIS [26]. In the last decade, the development of DCR has gone through a multi-disciplinary design process which has been constantly informed and evaluated by ethnographic and understandability studies [26].

The DCR language [27] consists of a core notation and a set of extensions. The core notation provides the basic constructs to create DCR models (i.e., also called *DCR graphs*). A DCR graph consists of nodes and directed edges. Nodes represent process activities, and edges represent relations defining the interplay between the process activities. Unconstrained activities can be executed at any time and any number of times. Likewise, blocks of interrelated activities, forming *a weakly connected component*[5] within the graph, can be executed with no influence from other (if any) weakly connected components in the graph.

In the DCR Language, activities have a marking (state) composed of three Boolean values: *executed*, *included* and *pending*. The *executed* marking specifies whether an activity has already been executed. The *included* marking indicates the relevance of an activity for the process at a certain stage of execution. Relevant activities have an included marking state and can constrain the execution of other activities, whereas irrelevant activities cannot. The *pending* marking, in turn, is used to set the requirement that an activity must be executed before the process ends.

There are six relations in the DCR language. In the following, we use the terminology "source" and "destination" to refer to a pair of activities linked with a DCR relation represented as a directed edge from a source activity to a destination activity. A *condition* denotes that a destination activity cannot be executed before a source activity is executed at least once in the past. A *milestone* denotes that a destination activity cannot be executed while a source activity is required (i.e., has the *pending* marking). The *inclusion* and *exclusion* relations denote that a source activity can make a destination activity, respectively, *relevant* or *irrelevant* for the process by switching its *included* marking. Finally, *response* and *no-response* relations denote that a source activity can make a destination activity, respectively, *required* or *no longer required* in the process by toggling its *pending* marking [2]. If a pair of activities is linked with more than a single relation, the effects of individual relations are combined and

---

[4] see `https://www.dcrgraphs.net` and `https://dcrsolutions.net`

[5] A weakly connected component is a maximal sub-graph where all nodes are connected by some path, ignoring the direction of the edges.

applied simultaneously. For instance, *include* and *response* relations between a pair of activities would denote that a source activity can make a destination activity relevant *and* whenever the source activity is executed, the destination activity becomes required.

The complexity metrics presented in Section 3 are based on this core notation.

2.3 Complexity Metrics

The complexity metrics proposed in the process modeling literature take roots in different fields including graph theory, software engineering and information theory [7,40]. In graph theory, several metrics have been adapted to estimate the complexity of process models [7,40]. These metrics, for instance, address factors like graph density, connectivity and degree of vertices [63]. Similarly, from software engineering, notable metrics have been adjusted to measure the complexity of process models. Metrics such as Lines of Code (LOC) [29], McCabe Cyclomatic Complexity [38, 39], Halstead's measures [24], and Henry and Kafura information flow metric [25] have been reformulated based on a pre-defined analogy made between source-code and process model constructs [7]. In addition, other metrics have been inspired by Shao and Wang's cognitive framework [65]. In this class of metrics, cognitive weights are assigned to different control-flow structures (e.g., sequence, branching, iteration) depending on their presumed complexity. With regards to information theory, Shannon entropy [64] provides a measure for randomness, variability and uncertainty in the information. This measure has been adjusted to quantify, for instance, the structuredness [9] and heterogeneity (i.e., diversity) of process model constructs [40].

The aforementioned metrics served as a basis for the development of a wide range of complexity metrics for process models. A recent literature review [36] counted more than 200 metrics, nearly half of which were empirically evaluated. The proposed metrics address the complexity of models in imperative languages like Business Process Modeling Notation (BPMN) [49], EPC [30], Business Process Execution Language (BPEL) [28] and Yet Another Workflow Language (YAWL) [1]. When it comes to declarative languages, to the best of our knowledge, the only existing study [35] investigates the complexity of process models expressed in the Case Management Modeling and Notation (CMMN) language [48]. Therein, the author proposed 3 metrics addressing the size, depth (i.e., number of nests in the models) and cognitive complexity of CMMN models (i.e., based on cognitive weights assigned subjectively to different CMMN elements). However, the results of an empirical study [36] testing the impact of the factors, captured by these metrics, on perceived difficulty and comprehension of CMMN models have failed to provide any statistical evidence supporting these metrics.

In this paper, we advance the existing state-of-the-art and build upon existing metrics to derive new variants that can be used to measure the complexity of declarative process models, particularly those formulated in the DCR language.

3 Metrics for Declarative Process Models

This work aims at providing complexity measures for declarative process models. To this end, we build upon (a) a recent study where the aspects defining the quality of declarative process have been identified [3] and (b) a comprehensive study proposing a set of complexity metrics for imperative process models in Event-driven Process Chain (EPC) [40] language. We contrasted the quality aspects identified in [3] with the metrics proposed in [40]. From there, as mentioned in Section 1, we were able to organize the existing metrics into two classes: a class of metrics emphasizing the complexity of the sequence-flow of imperative process models and thus cannot be applied to declarative models (e.g., concurrency) and another class of metrics capturing the graph structure of the model which could be potentially shared with declarative process models (e.g., density). The metrics in the latter class were further explored in the literature and used as a basis for deriving a new set of metrics that can be applied to declarative process models, particularly in the DCR language. All in all, we converged to the following four metrics: *size*, *density*, *separability* and *relation variability*.

In the following sections (Sections 3.1–3.4) we present the theoretical foundations and definitions underlying these metrics. In the formal definitions, the set of nodes (*activities*) of a DCR graph G (cf. Section 2.2) is denoted by A_G, and the set of its *relations* is denoted by R_G; and for some set X, its cardinality is denoted by $|X|$.

3.1 Size

Theoretical Foundations. The size metric is similar the LOC metric proposed in software engineering and adapted in process modeling to derive a broad range of language-specific metrics [7, 12, 22, 43, 59, 61].

The assumption behind using this metric is that the more elements a declarative process model has, the more difficult it is for a reader to understand. This assumption can be supported by the CLT positing that humans have limited cognitive abilities, that get challenged with increased information intake and processing [8, 42, 71]. Nevertheless, we recognize that interpreting constraints in a declarative model may require increased cognitive load as compared to activities. This factor, however, is covered by the density metric introduced in the next section (cf. Section 3.2).

Metric Definition. The size metric denotes the sum of activities and relations in the model. Since $|A_G|$ and $|R_G|$, respectively, denote the numbers of *activities* and *relations* of a DCR graph G, the size of Graph G can be defined as:

$$S(G) = |A_G| + |R_G|$$

3.2 Density

Theoretical Foundations. The density metric can be attributed to graph theory. Nevertheless, several variants of this metric have been adapted in process modeling research [40].

In declarative process models, an increased number of constraints for the same number of activities, within a weakly connected component, is expected to challenge readers' in understanding the encoded control-flow. This is due to the high coupling between the activities, which, in turn, requires more checks to evaluate the influence of each activity on the rest of activities in the component. We conjecture that performing all these checks would involve more mental resources, which will induce a higher cognitive load. This assumption can be supported by the limited computational offloading ability [62] of declarative process models (compared to imperative ones) as readers need to mentally compute the interplay imposed by the model constraints in order to evaluate whether an execution path is allowed in the model [81]. Increasing the number of constraints, in turn, would require more processing in the working memory and thus increase readers' cognitive load.

Metric Definition. The density metric denotes the maximum ratio of relations to activities in the *weakly connected components*[5] of the graph (cf. Section 2.2). Let *Comp*(G) be the set of weakly connected components of G: $\{c1, \ldots, c_n\}$, and let A_c and R_c be, respectively, the activities and relations in the weakly connected component $c \in Comp(G)$, then the density of Graph G can be defined as:

$$D(G) = \max_{c \in Comp(G)} \frac{|R_c|}{|A_c|}$$

3.3 Separability

Theoretical Foundations. The separability metric can be associated with the coupling and cohesion metrics, which describe the degree of interdependence between pairs of software modules [77].

In declarative process models, sets of activities that are within the same weakly connected component can be executed without considering the rest of the graph (cf. Section 2.2). This means that the checking of constraints becomes much easier since each component can be treated separately from the others. Based on this idea, we assume that the more weakly connected components there are in a declarative process model, the easier it is to comprehend. This assumption finds support in cognitive psychology, particularly with respect to the concept of *chunking*, which refers to a cognitive process used by humans to retain and process information [21, 45, 72]. During this process, information is divided into small pieces (i.e., chunks) and then integrated into a meaningful whole [21, 45, 72]. In the literature, it is claimed that the way information is sliced in the visual representation influences the mental effort required to retain it [20, 60, 70, 78]. For instance, phone numbers (e.g., 061093043) are easier to retain when digits are separated by dashes (e.g., 061-093-043) [78]. This is because the chunks are already explicit in the textual representation, which saves the reader the effort that would have been required to find the appropriate strategy to slice the information into chunks. Similarly, when dealing with declarative process models, components can be seen as chunks that are explicitly represented in the model. These chunks refer to distinct parts of the model that can be retained and processed in a detached manner with a reduced cognitive load.

Metric Definition. The separability metric denotes the ratio of weakly connected components to the size of the model. The separability of a graph G can be defined as:

$$E(G) = \frac{|Comp(G)|}{|A_G| + |R_G|}$$

3.4 Relation Variability

Theoretical Foundations. The relation variability metric originates from Shannon entropy, which has been proposed in Information Theory [64]. The metric has been used to capture a plenitude of natural and social phenomena where randomness, variability and uncertainty are key features [11, 15, 33, 52, 74, 79]. Similarly, Shannon entropy was adapted to measure the complexity of imperative process models [9, 40]. For instance, in [40], the metric was used to compute connector variability (heterogeneity) in EPC models, which in more general terms, denotes the extent to which different types of control-flow structures are incorporated in the model [40].

In a similar way, Shannon entropy can be applied to declarative process models to quantify the extent to which different types of constraints are used in the model. We assume that the more constraint types are incorporated in a model, the more difficult it is to comprehend. Indeed, understanding textual or visual representations requires interpreting the semantics encoded by the underlying language constructs, which in turn, requires humans to retrieve, from their long-term memory, the previously acquired concepts and rules allowing to perceive the encoded semantics. Therein, having a representation incorporating many language constructs with different semantics may require humans to retrieve a large amount of concepts and rules (from their long-term memory to their working memory) and keep switching between them to develop an overreaching understand of the representation in hand, which would pose an increased cognitive load on their working memory [10]. The same effect is expected to occur when dealing with declarative process models incorporating constraints with different semantics.

Metric Definition. The relation variability metric denotes the maximum entropy over the different *relation types* in the components of the model (cf. Section 2.2).

Let $\mathcal{T} = \{\mathtt{c}, \mathtt{m}, \mathtt{i}, \mathtt{e}, \mathtt{r}, \mathtt{n}\}$ be the set of the different types of DCR Relations (i.e., `c`onditions, `m`ilestones, `i`ncludes, `e`xcludes, `r`esponses, `n`o-responses). Let $\mathcal{T}_c$ denote the set of relation types observed in a component c (part of a DCR graph G) and let R_c^t denote the relations with type t within a component c (part of a DCR graph G). Similarly to [40], given a component c and a relation type t, the entropy is based of the relative frequency p:

$$p(c,t) = \begin{cases} \dfrac{|R_c^t|}{|R_c|} & \text{if } |R_c| > 0 \\ 0 & \text{otherwise} \end{cases}$$

The actual entropy is the negative sum over the six relation types of $p(c,t)\log_6(p(c,t))$. Note that, similarly to [40], the base of the log function corresponds to the number of relation types in the language. Based on that, the relation variability of a DCR graph G can be defined as:

$$V(G) = \max_{c \in Comp(G)} \left\{ -\sum_{t \in \mathcal{T}_c} p(c,t)\log_6(p(c,t)) \right\}$$

Please note that this metric assumes that the DCR model contains at least one relation.

4 Design of Empirical Study

This section describes the research method followed to design and conduct the experiment. Section 4.1 introduces the research model and the hypotheses. Section 4.2 describes the experiment materials. Section 4.3 provides an overview about the participants recruited in the experiment. Section 4.4 describes the experiment design and procedure. Section 4.5 outlines the data analysis approach.

4.1 Research Model and Hypotheses

This section presents the research model and the hypotheses guiding the empirical study.

The theoretical foundations presented in Section 3 suggest that the factors captured by the proposed complexity metrics (i.e., size, density, separability, relation variability) impact users' cognitive load (which we estimate in terms of perceived difficulty, comprehension accuracy and answering time [8]). This empirical study aims at providing evidence to support this proposition. Overall, we test 12 hypotheses, devised into 4 sets, each addressing a particular metric. The hypotheses can be formulated as follows:

Size Metric. $\mathbf{H1_a}$: Declarative process models with increased size are *perceived more difficult* than declarative process models with reduced size. **$\mathbf{H1_b}$:** Declarative process models with increased size have *lower comprehension accuracy* than declarative process models with reduced size. **H1$_c$:** Declarative process models with increased size require *more time* to be read than declarative process models with reduced size.

Density Metric. H2$_a$: Declarative process models with increased density are *perceived more difficult* than declarative process models with reduced density. **H2$_b$:** Declarative process models with increased density have *lower comprehension accuracy* than declarative process models with reduced density. **H2$_c$:** Declarative process models with increased density require *more time* to be read than declarative process models with reduced density.

Separability. H3$_a$: Declarative process models with reduced separability are *perceived more difficult* than declarative process models with increased separability. **H3$_b$:** Declarative process models with reduced separability have *lower comprehension accuracy* than declarative process models with increased separability. **H3$_c$:** Declarative process models with reduced separability require *more time* to be read than declarative process models with increased separability.

Relation Variability. H4$_a$: Declarative process models with increased relation variability are *perceived more difficult* than declarative process models with reduced relation variability. **H4$_b$:** Declarative process models with increased relation variability have *lower comprehension accuracy* than declarative process models with reduced relation variability. **H4$_c$:** Declarative process models with increased relation variability require *more time* to be read than declarative process models with reduced relation variability.

The empirical study is designed as a controlled experiment [76]. Our research model is depicted in Figure 1. In the treatment side, the factors captured by the proposed metric denote the investigated theoretical constructs (**T**). Each of these constructs is operationalized (**O**) into two levels: a “reduced (low) level” and an “increased (high) level”. These two levels are tested on process models on which the metric, capturing the addressed factor returns a low or a high value respectively (cf. Sections 4.2).

As for the output, the investigated factors are expected to impact perceived difficulty, comprehension accuracy and response time. Perceived difficulty is operationalized using a 6-point likert scale (1: very easy, 6: very difficult, similar to [44]) on which participants can self-assess the difficulty they perceived when solving a particular task. Comprehension accuracy, in turn, is operationalized as answer correctness that is a binary measure capturing whether a task is answered correctly or not. Lastly, response time is operationalized as answering time that is the time elapsed when solving a particular task.

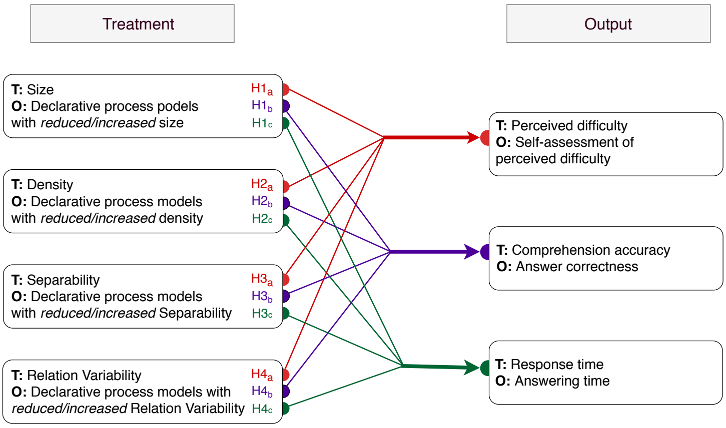
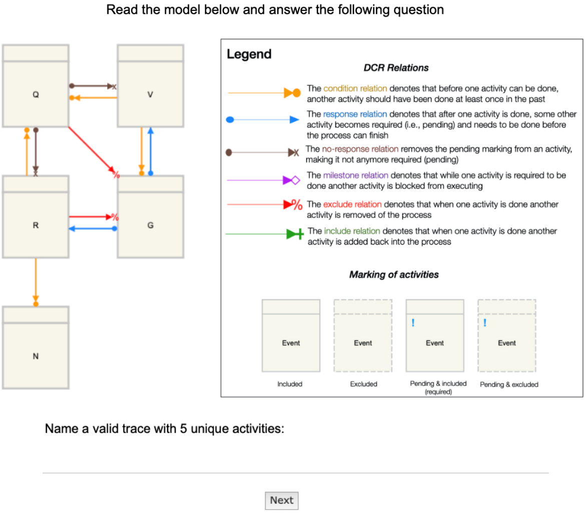


Fig. 1: Research Model. Abbreviations: T: Theoretical Construct, O: Operationalization of Construct.

4.2 Materials

The material designed for this experiment comprises declarative process models in the DCR language and a set of tasks addressing the comprehension of these models.

The process models are designed to reflect the levels of the different factors introduced in Section 4.1. However, the models might also comprise confounding factors, which can threaten the validity of the experiment. To address this issue, we have identified a set of (model-related) confounding factors which we have mitigated during the design of the experiment. Following the guidelines in [80] and the declarative modeling practices identified in [2], we denote (a) *layout*, (b) *modeling constructs*, (c) *relation patterns* and (d) *process scenario* as 4 pertinent aspects which can confound the results of the experiment. We address the layout factor *(a)* by defining a uniform and carefully set up layout, ensuring a proper spacing between the model activities and reducing crossing arrows (relations) as much as possible. Regarding the modeling constructs *(b)*, we rely only on the basic DCR notation (cf. Section 2.2). Hence, the designed models do not incorporate advanced concepts such as sub-processes [14] or contextual process data [68], which could induce additional (confounding) factors to the experiment. As for the modeling patterns *(c)*, we limit the design of the models to a pre-defined set of relation patterns, which comprises the 6 individual relations of the DCR notion (cf. Section 2.2) and some combinations of them. These patterns were explained and illustrated to the participants in individual tutorial sessions which have taken place prior to the experiment (cf. Section 4.4). Regarding the process scenario *(d)*, domain-knowledge can influence the participants' understanding of the process [3,80]. To mitigate the effect of this factor, we anonymize the model activities by assigning them labels corresponding to random alphabet letters (similar to [55]).



Read the model below and answer the following question

Legend

DCR Relations

The condition relation denotes that before one activity can be done, another activity should have been done at least once in the past

The response relation denotes that after one activity is done, some other activity becomes required (i.e., pending) and needs to be done before the process can finish

The no-response relation removes the pending marking from an activity, making it not anymore required (pending)

The milestone relation denotes that while one activity is required to be done another activity is blocked from executing

The exclude relation denotes that when one activity is done another activity is removed of the process

The include relation denotes that when one activity is done another activity is added back into the process

Marking of activities

Name a valid trace with 5 unique activities:

Fig. 2: Example of the experiment tasks.

Regarding the design of the tasks, each task comprises a declarative process model, an inference question and a legend describing the semantics of the core DCR notation (including relations and activity markings, cf. Section 2.2). An example of the experiment tasks is shown in Figure 2. The inference questions prompt participants to name a valid execution path (i.e., trace) where all the model activities are executed once and only once. The choice of this question type is driven by several motivations. Firstly, unlike dichotomous questions (i.e., true or false), the chances of correctly guessing the answer to inference questions are very low. Secondly, the investigated metrics capture features spanning across the entire model. Hence, to capture their effects, it is required to provide questions making the entire model relevant to the question. As explained in [54], when dealing with process models, users do not focus on the entire model, but rather limit their attention to the task-relevant parts of the model. In our context, participants' performance on questions addressing a subset of the model would not necessarily reflect the complexity of the overall model that is captured by the proposed metrics. Therefore, to ensure that the investigated factors are well captured by the questions, it is crucial to provide questions requiring participants to focus on the entire model. Last but not least, the chosen question type requires to perceive the execution order, the dependencies between activities and their influence on each other. These aspects are considered important when trying to interpret the control-flow encoded in declarative process models [81].

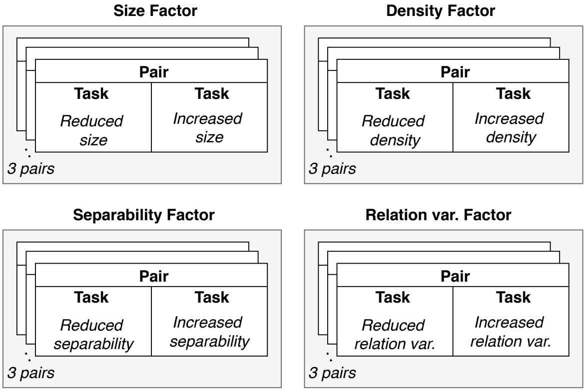


Fig. 3: Organization of Experiment Tasks.

The experiment comprises 24 tasks that are organized as shown in Figure 3. Therein, tasks (and the underlying models) are distributed into 4 sets, each set addressing a particular factor (i.e., size, density, separability, relation variability). Then, within each set, tasks are grouped into pairs. Each pair, in turn, comprises two tasks addressing respectively, the reduced and the increased levels of the factor (e.g., reduced size, increased size). As each factor is tested separately, the process models within each pair of tasks are equal on all the covered metrics, except for the one controlled by the levels of the factor under investigation. To obtain more data-points per participant, we provide 3 task pairs for each factor. At the analysis, a pairwise comparison approach is used to compare the factor levels within each pair of tasks (cf. Section 4.5). The process models used for the experiment (together with the computed metrics) are available online[6].

4.3 Participants

The experiment covers 16 participants recruited from an academic environment (e.g., students, professors). These participants have different levels of expertise in process modeling using DCR. On a 7-points likert scale asking participant

---

[6] See `http://andaloussi.org/MetricsPaper2020/`

to rate their familiarity with the DCR language (1: unfamiliar, 7: very familiar), half answered in the range [1,4], while the other half answered in the range [5,7]. The effects associated with differences in participants' expertise are addressed in the experiment design (cf. Section 4.4).

4.4 Experiment Design and Procedure

The experiment is designed following a within-subject approach. Herein, each participant is exposed to all factors and factor levels several times. This design allows repeated measurements, which in turn provide more precise data as each participant generates data-points for every factor level [23]. Moreover, the choice of this design allows to deal with the heterogeneity of participants' background that is another confounding factor which we mitigate in our experiment [23].

The experiment is conducted in a controlled lab-environment. Figure 4 illustrates the experiment procedure, which is performed in individual sessions. A participant is invited to a tutorial where he is familiarized with the core DCR notation, the used relation patterns and the type of tasks provided in the experiment. Following that, the participant is given a quiz including four comprehension tasks (with different levels of complexity) to evaluate his understanding of the covered material. Then, the quiz answers are reviewed and discussed to help the participant identify his mistakes and improve his understanding of the DCR notation. These two steps are important to ensure that all participants (regardless of their expertise) have the baseline knowledge required for the experiment. Afterwards, a survey is administered to collect demographic information about the participant's expertise in declarative process modeling using DCR (cf. Section 4.3). The (main) experiment, in turn, comprises a series of comprehension tasks designed following the approach explained in Section 4.2. The tasks did not involve any time restrictions. To address a potential learning effect, the pairs and sets of tasks are presented in different orders to each participant. Each task is followed by a 6-point likert scale asking the participant to assess the difficulty perceived when solving that task. By the end of the experiment, the participant is invited to an interview to reflect about his experience when answering the different tasks.

From the experiment, we collected participants' answers, answering time and self-assessments of perceived difficulty. In addition we collected verbal data (recorded from the interviews) and physiological data (collected from eye-tracking and galvanic skin response devices) which we are planning to analyze in a follow-up study.

4.5 Data Analysis

The data considered for the analysis contains participants' self-assessment of perceived difficulty, answering time, and answers correctness (obtained by assigning the participants' answers a binary score depending on their correctness). From 16 participants, we obtained 48 data-points per measure/factor level.

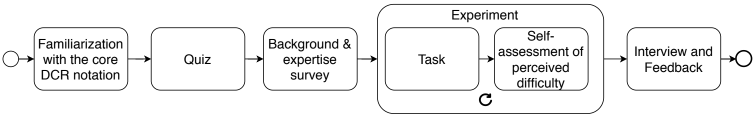


Fig. 4: Experiment Procedure.

At the analysis, we used a pairwise approach to group and compare the two levels within of each the 4 factors. These comparisons were done in terms of the 3 collected measures (perceived difficulty, answering time, answers correctness). In total, we analyzed 12 paired data samples (see the concatenation of the columns: *Factor level* and *Measure* in Table 1). We used the Shapiro-Wilk Test, quantile-quantile and histogram Plots to investigate the normality in each paired sample. The results show that the data in 11 (out of 12) paired samples is not normally distributed. For this data, we administered the Wilcoxon signed-rank Test to investigate the differences within paired samples. Regarding the remaining paired sample (factor: size, measure: answer correctness), the variance of one data sample in the pair was equal to 0. Therein we conducted the Shapiro-Wilk Test on the other sample (of the pair) and based on that, we used the Wilcoxon signed-rank Test as the data was not normally distributed in that single sample.

In addition, we computed descriptive statistics for the different factor levels. The results of the hypothesis-testing and the descriptive statistics are reported in Section 5.

5 Findings

This section presents the findings of the study. Sections 5.1, 5.2, 5.3, 5.4 report the descriptive statistics and the results of the inferential tests addressing, respectively, the hypotheses about the size, density, separability and relation variability as defined in Section 4.1. Summaries of the descriptive statistics and inferential tests are provided in Tables 1, 2 respectively.

5.1 Size

Descriptive Analysis. The descriptive statistics (cf. Table 1) *suggest* that models with increased size are perceived more difficult, have lower comprehension accuracy and require more time to be read, which in turn, hint that large models are associated with increased cognitive load. More specifically, the average (self-assessment of) perceived difficulty (measured in the range [1: very easy, 6: very difficult]) was higher in large models (M=4.271) compared to small models (M=1.854) (i.e., $\mathbf{H1_a}$). Conversely, the average answer correctness (measured in range [0,1]) was lower in large models (M=0.833) compared to small models (M=1) (i.e., $\mathbf{H1_b}$). The average answering time (in seconds), in turn, was higher in large models (M=153.349) compared to small models

Factor	Factor level	Measure	N	Min.	Max.	M	SD
Size	Reduced size	Perceived diff.	48	1	4	1.854	0.85
	Increased size		48	2	6	4.271	1.047
	Reduced size	Answer correct.	48	1	1	1	0
	Increased size		48	0	1	0.833	0.377
	Reduced size	Answering time	48	10.76	85.21	30.91	15.637
	Increased size		48	41.64	332.97	153.349	68.073
Density	Reduced density	Perceived diff.	48	1	4	2.396	0.765
	Increased density		48	2	6	3.979	1.021
	Reduced density	Answer correct.	48	0	1	0.792	0.41
	Increased density		48	0	1	0.75	0.438
	Reduced density	Answering time	48	19	127.27	56.583	26.326
	Increased density		48	41.8	434.3	122.702	85.22
Separability	Reduced separability	Perceived diff.	48	2	6	3.104	1.036
	Increased separability		48	1	4	1.563	0.681
	Reduced separability	Answer correct.	48	0	1	0.875	0.334
	Increased separability		48	0	1	0.938	0.245
	Reduced separability	Answering time	48	29.42	225.52	88.713	48.726
	Increased separability		48	14.87	115.51	37.903	22.141
Relation var.	Reduced relation var.	Perceived diff.	48	1	4	1.917	0.846
	Increased relation var.		48	2	5	3.188	0.891
	Reduced relation var.	Answer correct.	48	0	1	0.938	0.245
	Increased relation var.		48	0	1	0.708	0.459
	Reduced relation var.	Answering time	48	16.45	137.69	52.109	28.598
	Increased relation var.		48	33.11	229.59	84.663	42.157

Table 1: Descriptive statistics. Abbreviations: Min.: M: median, SD: standard deviation, Perceived diff.: self-assessment of perceived difficulty, correct.: correctness, var.: variability. Notes: The measures: self-assessment of perceived difficulty, answer correctness, answering time represent, respectively, the operationalization of the constructs: perceived difficulty, comprehension accuracy and response time (cf. Figure 1). The unit for answering time is second.

H.	Factor	Measure	W	p
$\mathbf{H1_a}$	Size	Perceived diff.	0.000	<**.001**
$\mathbf{H1_b}$		Answer correct.	0.000	<**.001**
$\mathbf{H1_c}$		Answering time	0.000	<**.001**
$\mathbf{H2_a}$	Density	Perceived diff.	0.000	<**.001**
$\mathbf{H2_b}$		Answer correct.	33.000	0.565
$\mathbf{H2_c}$		Answering time	0.000	<**.001**
$\mathbf{H3_a}$	Separability	Perceived diff.	0.000	<**.001**
$\mathbf{H3_b}$		Answer correct.	20.000	0.299
$\mathbf{H3_c}$		Answering time	11.000	<**.001**
$\mathbf{H4_a}$	Relation var.	Perceived diff.	54.000	<**.001**
$\mathbf{H4_b}$		Answer correct.	104.000	**0.005**
$\mathbf{H4_c}$		Answering time	69.000	<**.001**

Table 2: Results of inferential tests addressing the hypotheses defined in Section 4.1, obtained using Wilcoxon Signed–Rank Test. Abbreviations: Perceived diff.: self-assessment of perceived difficulty, correct.: correctness, var.: variability. Note: The measures: self-assessment of perceived difficulty, answer correctness, answering time represent, respectively, the operationalization of the constructs: perceived difficulty, comprehension accuracy and response time (cf. Figure 1).

(M=30.91) (i.e., $\mathbf{H1_c}$). In the next paragraph, we turn towards inferential statistics to test the significance of these differences and thus validate our hypotheses.

Hypotheses Testing. The results of the inferential tests (cf. Table 2) show that declarative process models with increased size are perceived more difficult (W=0, p=< .001), have lower comprehension accuracy (V=0, p=< .001) and require more time to be read (W=0, p=< .001) than those with reduced size, which in turn support, respectively, Hypotheses $\mathbf{H1_a}$, $\mathbf{H1_b}$ and $\mathbf{H1_c}$.

5.2 Density

Descriptive Analysis. The descriptive statistics (cf. Table 1) *suggest* that models with increased density are perceived more difficult, have lower comprehension accuracy and require more time to be read, which hint that dense models are associated with increased cognitive load. Indeed, the average perceived difficulty was higher in dense models (M=3.979) compared to less dense models (M=2.396) (i.e., $\mathbf{H2_a}$). Conversely, the average answer correctness was lower in dense models (M=0.75) compared to the less dense ones (M=0.792) (i.e., $\mathbf{H2_b}$). The average answering time, in turn, was higher in dense models (M=122.702) compared to less dense models (M=56.583) (i.e., $\mathbf{H2_c}$). These differences are investigated using inferential statistics in the next paragraph.

Hypotheses Testing. The results of the inferential tests (cf. Table 2) show that declarative process models with increased density are perceived more difficult (W=0, p=< .001) and require more time to be read (W=0, p=< .001) than those with reduced density. Hence, Hypotheses $\mathbf{H2_a}$ and $\mathbf{H2_c}$ are confirmed. However, $\mathbf{H2_b}$ cannot be confirmed as the difference is not significant (W=33, p=0.565). Therefore, the claim that dense models have lower comprehension accuracy cannot be supported.

5.3 Separability

Descriptive Analysis. The descriptive statistics (cf. Table 1) *suggest* that models with reduced separability are perceived more difficult, have lower comprehension accuracy and require more time to be read, which hint that models with reduced separability are associated with an increased cognitive load. More specifically, the average perceived difficulty was higher in models with reduced separability (M=3.104) compared to those with increased separability (M=1.563) (i.e., $\mathbf{H3_a}$). Conversely, the average answer correctness was lower in models with reduced separability (M=0.875) compared to models with increased separability (M=0.938) (i.e., $\mathbf{H3_b}$). The average answering time, in turn, was higher in models with reduced separability (M=88.713) compared to those with increased separability (M=37.903) (i.e., $\mathbf{H3_c}$). In the next paragraph, we use inferential statistics to investigate these differences.

Hypotheses Testing. The results of the inferential tests (cf. Table 2) show that declarative process models with reduced separability are perceived more difficult (W=0, p=< .001) and require more time to be read (W=11, p=< .001) than those with increased separability, which provide empirical evidence for Hypotheses $\mathbf{H3_a}$ and $\mathbf{H3_c}$. However, likewise the density factor, the inferential statistics do not support $\mathbf{H3_b}$ (W=20, p=0.299). Hence, we cannot posit that models with reduced separability have lower comprehension accuracy.

5.4 Relation Variability

Descriptive Analysis. The descriptive statistics (cf. Table 1) *suggest* that models with increased relation variability are perceived more difficult, have lower comprehension accuracy and require more time to be read, which hint that models with increased relation variability are associated with higher cognitive load. In particular, the average perceived difficulty was higher in models with increased relation variability (M=3.188) compared to those with reduced relation variability (M=1.917) (i.e., $\mathbf{H4_a}$). Conversely, the average answer correctness was lower in models with increased relation variability (M=0.708) compared to models with reduced relation variability (M=0.938) (i.e., $\mathbf{H4_b}$). The average answering time, in turn, was higher in models with increased relation variability (M=84.663) compared to those with reduced variability (M=52.109) (i.e., $\mathbf{H4_c}$). Inferential statistics are used in the next paragraph to investigate these differences.

Hypotheses Testing. The results of the inferential tests (cf. Table 2) show that declarative process models with increased relation variability are perceived more difficult (W=54, p=< .001), have lower comprehension accuracy (W=104, p=0.005) and require more time to be read (W=69, p=< .001), which in turn provide evidence for all the hypotheses related the relation variability factor i.e.,$\mathbf{H4_a}$, $\mathbf{H4_b}$ and $\mathbf{H4_c}$ respectively.

6 Discussion

This section discusses the findings of this work. Overall, the results of the inference tests support 10 (out of 12) hypotheses, which in turn, suggest that the factors captured by the size, density, separability and relation variability metrics influence users' cognitive load (in terms of the used measures, cf. Section 7 for a related threat to construct validity). Nevertheless, two hypotheses (i.e., $\mathbf{H2_b}$ and $\mathbf{H3_b}$) could not be verified. Namely, we could not show that the density and separability factors impact participants' comprehension accuracy. The findings of these two hypotheses can be subject to many interpretations. Regarding $\mathbf{H2_b}$ (impact of density on comprehension accuracy), we think that the design of the experiment contributed to this result. As mentioned in Section 4.4, the experiment's tasks did not involve any time restrictions. Hence, the participants might have had enough time to focus on the difficult tasks involving dense models (as confirmed by $\mathbf{H2_c}$) and thus, they could score similarly to tasks incorporating models with reduced density.

Regarding $\mathbf{H3_b}$ (impact of separability of comprehension accuracy), a possible explanation for the lack of significant differences could be associated with the ceiling effect, which typically occurs when most of the participants achieve high scores [75]. Indeed, the average answer correctness on tasks incorporating complex models with reduced separability is very high (M=0.875). It is also the highest score among the other tasks incorporating complex models (with increased size, increased density, or increased relation variability). Nevertheless, as confirmed in $\mathbf{H3_a}$ and $\mathbf{H3_c}$ models with reduced separability are still perceived difficult and require more time to be read. A possible interpretation of this effect could be that although participants might have experienced an increased level of cognitive load, they didn't reach the state of cognitive overload [8]. This proposition is supported by Veltman and Jansen [73], who posit that although users' mental effort increases when experiencing high cognitive load, their performance remains stable as long as they do not reach a state of cognitive overload. This assumption can also explain the significant difference in perceived difficulty between the two factor levels since participants had to invest more mental resources in the models with reduced separability. Nevertheless, users' performance can also be estimated in terms of response time [8], which was significantly different between the two factor levels. This in turn, hints towards a third explanation that is related to the unrestricted task time which is similar to the explanation suggested for $\mathbf{H2_b}$.

The outcome of this research contributes to a better understanding of complexity in declarative process modeling. As mentioned in Section 1, the proposed metrics can be used to evaluate the design of existing declarative process models and decide on whether a structural change is required to reduce the complexity of the model and improve its understandability. In addition, the proposed metrics can be implemented by tool vendors to automate and enable the evaluation of declarative process models at design time. At first, the metrics can give indications about which models are complex, then with further research, these tools can suggest structural changes to optimize the metrics while preserving the original model behavior. Such optimization is be based on the assumption that a process can be modeled differently using various modeling patterns. In addition, the proposed metrics can serve as heuristics to support existing declarative process mining algorithms [4,46,66] and guide the discovery of models with reduced complexity.

7 Threats to Validity

Our experiment can be subject to several threats to validity, which we discuss in the following paragraphs.

Internal Validity. Threats to internal validity cover the effects that can threaten the causal relationship between the independent and dependent variables [76]. In that regard, the lack of instructions and control over the external environment can affect the results of the experiment. To mitigate this threat, the experiment was conducted in a controlled environment and participants were uniformly instructed about the experiment procedure following a pre-defined protocol.

The design of the experiment could also affect the internal validity of the study. With regards to the used models, we identified and mitigated the confounding factors susceptible to impact participants' understanding of the model (cf. Section 4.2). As for the tasks, we formulated the comprehension questions in the same way to ensure no disparities in that respect (cf. Section 4.2). Regarding the participants, we recognize that our sample is heterogeneous and thus some participants were more experienced than others. However, we provided a uniform familiarization to all the participants ensuring they all have the basic knowledge to participate in the study. In addition, we used a within-subject design and a pairwise comparison approach to mitigate this effect (cf. Sections 4.4 and 4.5). Another threat is associated with a potential learning effect during the experiment. Indeed, the more tasks participants perform, the more experience they acquire. To mitigate this effect, the material was provided in a randomized order. Therefore the learning effect was uniformly spread out over the different factor levels.

External Validity. Threats to external validity emphasize the limited ability to generalize the experiment results to industrial settings [76]. Therein, a number of threats can be identified. Firstly, the used process models were represented in the DCR language. Therefore, the results cannot be generalized to all process modeling languages. However, given the synergies between DCR and other declarative languages (e.g., Declare [53]), we presume that some of the findings could potentially apply to other languages in the declarative paradigm. Moreover, as the experiment covered the core DCR notation, our findings cannot be generalized to DCR models, including advanced concepts (e.g., sub-processes, contextual annotations). Our focus on the core DCR notation was intentional at this stage of research as the use of advanced concepts may induce confounding factors that can hinder the internal validity of the study. In addition, the anonymized labels applied to the model activities (cf. Section 4.2) do not reflect the way process models are communicated in the real-world. Nevertheless, we support our design decision by the need to mitigate the influence of domain-knowledge (about the process scenario) on the understandability of the model, which otherwise can affect the internal validity of the experiment.

Regarding the design of the tasks (cf. Section 4.2), we recognize that the used task type is not representative of all possible uses of declarative process models. Nevertheless, following a series of meetings between the co-authors, we agreed that this task type is adequate for capturing the complexity of the entire process model, which is crucial for evaluating our complexity metrics. In addition, we presume that answering such a task requires a series of checks (e.g., activities' order, dependencies, influences on each other), which are deemed important when interpreting the control-flow encoded in declarative process models [81].

Last but not least, while the current empirical study shows the impact of the factors captured by the covered complexity metrics, it is worthwhile to highlight that the experiment was conducted in laboratory settings and therefore, more empirical studies are required in industrial settings to generalize the impact of the investigated factors to the real-world.

Construct Validity. Threats to construct validity cover the ability to generalize the experiment's findings to existing concepts and theories [76]. It that respect, the used measures (i.e., self-assessment of perceived difficulty, answer correctness, answering time) have been suggested as indicators of cognitive load [8]. However, these measures are not as objective and precise as physiological measures derived form eye-tracking or galvanic skin response. These two modalities will be covered in a more advanced data analysis planned as part of a follow-up study.

Conclusion Validity. Threats to conclusion validity relate to the ability to draw correct conclusions about the findings of the study [76]. Our sample size could represent a threat in that respect. Since our experiment was conducted in individual sessions within a controlled environment, we could not have a larger sample. However to minimize this threat, we have conducted repeated measurements and thus collected 48 data-points per factor level, which is higher than the typical 25-30 data-points required to perform inferential statistics. The findings presented in this work, in turn, motivate the design of further empirical studies testing the presented metrics on a large scale. Finally, since our metrics cover only a subset of complexity factors, we cannot make definite conclusions about the overall complexity of declarative process models.

8 Conclusion and Future Work

This work presents a set of complexity metrics providing quantifiable means to assess the understandability of declarative process models. The factors captured by these metrics have been investigated in an empirical study and the results have shown their impact on users' cognitive load.

As future work, we are planning to analyze the physiological data collected from eye-tracking and galvanic skin response. Moreover, we are planning to conduct a qualitative analysis of the verbal data collected from the interviews in order to understand participants' engagement with declarative process models during comprehension tasks and further discuss the results of the inferential statistical tests. Besides our envisaged research, there are further directions for future work. On a practical level, the proposed metrics can be implemented and embedded in existing declarative process modeling tools, while on a more theoretical level, further research can be conducted to propose structural changes allowing to optimize these metrics (cf. Section 6). On an empirical level, in turn, large scale studies can be performed to further investigate these metrics. The studies can be conducted with people in less controlled environments (e.g., classrooms, companies). In addition, similar to [40], the metrics can be applied directly to repositories of declarative process models (e.g., from the DCR Portal) to study the relationship between the proposed metrics and the quality issues in exiting models.

References

1. van der Aalst, W., ter Hofstede, A.: YAWL: yet another workflow language. Information Systems 30(4), 245–275 (jun 2005), `http://linkinghub.elsevier.com/retrieve/pii/S0306437904000304`
2. 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: International Conference on Business Process Management. pp. 417–434. Springer (2020)
3. Andaloussi, A.A., 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 pp. 1–28 (2020)
4. Back, C.O., Slaats, T., Hildebrandt, T.T., Marquard, M.: Discover: Accurate & efficient discovery of declarative process models. arXiv preprint arXiv:2005.10085 (2020)

5. Baddeley, A.D., Hitch, G.: Working memory. In: Psychology of learning and motivation, vol. 8, pp. 47–89. Elsevier (1974)
6. Becker, J., Rosemann, M., Von Uthmann, C.: Guidelines of business process modeling. In: Business process management, pp. 30–49. Springer (2000)
7. Cardoso, J., Mendling, J., Neumann, G., Reijers, H.A.: A discourse on complexity of process models. In: International Conference on Business Process Management. pp. 117–128. Springer (2006)
8. Chen, F., Zhou, J., Wang, Y., Yu, K., Arshad, S.Z., Khawaji, A., Conway, D.: Robust Multimodal Cognitive Load Measurement, pp. 13–32 (2016), `https://doi.org/10.1007/978-3-319-31700-7{\_}2`
9. Cheng, C.Y.: Complexity and usability models for business process analysis (2008)
10. Cowan, N.: Working memory capacity: Classic edition. Psychology press (2016)
11. Dancoff, S., Quastler, H.: The information content and error rate of living things. Essays on the use of information theory in biology (1953)
12. Debnath, N., Salgado, C., Peralta, M., Riesco, D., Montejano, G.: Optimization of the business process metrics definition according to the bpdm standard and its formal definition in ocl. In: ACS/IEEE International Conference on Computer Systems and Applications-AICCSA 2010. pp. 1–8. IEEE (2010)
13. Debois, S., Slaats, T.: The analysis of a real life declarative process. In: 2015 IEEE Symposium Series on Computational Intelligence. pp. 1374–1382. IEEE (2015)
14. Debois, S., Hildebrandt, T.T., Slaats, T.: Replication, refinement & reachability: complexity in dynamic condition-response graphs. Acta Informatica 55(6) (2018)
15. Delgado-Bonal, A., Martín-Torres, J.: Human vision is determined based on information theory. Scientific reports 6, 36038 (2016)
16. Dengler, F., Vrandečić, D.: Wiki-based maturing of process descriptions. In: International Conference on Business Process Management. pp. 313–328. Springer (2011)
17. Dumas, M., La Rosa, M., Mendling, J., Reijers, H.A.: Business process management. Springer (2013)
18. Durso, F.T., Nickerson, R.S., Dumais, S.T., Lewandowsky, S., Perfect, T.J.: Handbook of applied cognition. Wiley Online Library (2007)
19. 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: Proceedings of EMMSAD. pp. 353–366 (2009)
20. Gobet, F., Lane, P.C., Croker, S., Cheng, P.C., Jones, G., Oliver, I., Pine, J.M.: Chunking mechanisms in human learning. Trends in cognitive sciences 5(6), 236–243 (2001)
21. Gray, P.: Psychology. Worth Publishers (2006), `https://books.google.dk/books?id=o6m9AyzSbAkC`
22. Gruhn, V., Laue, R.: Approaches for business process model complexity metrics. In: Technologies for business information systems, pp. 13–24. Springer (2007)
23. Gulden, J., Burattin, A., Andaloussi, A.A., Weber, B.: From analytical purposes to data visualizations: a decision process guided by a conceptual framework and eye tracking. Software & Systems Modeling pp. 1–24 (2019)
24. Halstead, M.H., et al.: Elements of software science, vol. 7. Elsevier New York (1977)
25. Henry, S., Kafura, D.: Software structure metrics based on information flow. IEEE transactions on Software Engineering (5), 510–518 (1981)
26. Hildebrandt, T.T., Andaloussi, A.A., Christensen, L.R., Debois, S., Healy, N.P., López, H.A., Marquard, M., Møller, N.L., Petersen, A.C., Slaats, T., et al.: Ecoknow: Engineering effective, co-created and compliant adaptive case management

systems for knowledge workers. In: Proceedings of the International Conference on Software and System Processes. pp. 155–164 (2020)

27. Hildebrandt, T.T., Mukkamala, R.R.: Declarative Event-Based Workflow as Distributed Dynamic Condition Response Graphs. EPTCS 69, 59–73 (2011)
28. Jordan, D., Evdemon, J., Alves, A., Arkin, A., Askary, S., Barreto, C., Bloch, B., Curbera, F., Ford, M., Goland, Y., et al.: Web services business process execution language version 2.0. OASIS standard 11(120), 5 (2007)
29. Kalb, G.E.: Counting lines of code, confusions, conclusions, and recommendations. In: Briefing to the 3rd Annual REVIC User's Group Conference (1990)
30. Keller, G., Nüttgens, M., Scheer, A.W.: Semantische Prozessmodellierung auf der Grundlage "ereignisgesteuerter Prozessketten (EPK)". Institut für Wirtschaftsinformatik Saarbrücken: Veröffentlichungen des Instituts für Wirtschaftsinformatik, Inst. für Wirtschaftsinformatik
31. Krogstie, J.: Quality in Business Process Modeling. Springer, Cham (2016)
32. Krogstie, J., Sindre, G., Jørgensen, H.: Process models representing knowledge for action: a revised quality framework. European Journal of Information Systems 15(1), 91–102 (2006)
33. Lee, R., Jonathan, P., Ziman, P.: Pictish symbols revealed as a written language through application of shannon entropy. Proceedings of the royal society a: Mathematical, physical and engineering sciences 466(2121), 2545–2560 (2010)
34. Lin, T., Imamiya, A., Mao, X.: Using multiple data sources to get closer insights into user cost and task performance. Interacting with Computers 20(3), 364–374 (2008)
35. Marin, M.A., Lotriet, H., Van Der Poll, J.A.: Metrics for the case management modeling and notation (CMMN) specification. In: Proceedings of the 2015 Annual Research Conference on South African Institute of Computer Scientists and Information Technologists. pp. 1–10 (2015)
36. Marin, M.A.: Exploring complexity metrics for artifact-centric business process models. University of South Africa (South Africa) (2017)
37. Marquard, M., Shahzad, M., Slaats, T.: Web-Based Modelling and Collaborative Simulation of Declarative Processes. In: Business Process Management. pp. 209–225. Springer International Publishing, Cham (2016)
38. McCabe, T.J.: A complexity measure. IEEE Transactions on software Engineering (4), 308–320 (1976)
39. McCabe, T.J., Butler, C.W.: Design complexity measurement and testing. Communications of the ACM 32(12), 1415–1425 (1989)
40. Mendling, J.: Detection and prediction of errors in EPC business process models. Ph.D. thesis, Wirtschaftsuniversität Wien Vienna (2007)
41. Mendling, J., Reijers, H.A., van der Aalst, W.M.: Seven process modeling guidelines (7pmg). Information and Software Technology 52(2), 127–136 (2010)
42. Miller, G.A.: The magical number seven, plus or minus two: Some limits on our capacity for processing information. Psychological review 63(2), 81 (1956)
43. Muketha, G.M., Abd Ghani, A., Selamat, M., Atan, R.: A survey of business processes complexity metrics (2010)
44. Müller, S.C., Fritz, T.: Using (bio) metrics to predict code quality online. In: 2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE). pp. 452–463. IEEE (2016)
45. Neath, I., Surprenant, A.: Human Memory: An Introduction to Research, Data, and Theory. Thomson/Wadsworth (2003), `https://books.google.dk/books?id=xZ9WAAAAYAAJ`

46. Nekrasaite, V., Parli, A.T., Back, C.O., Slaats, T.: Discovering responsibilities with dynamic condition response graphs. In: Giorgini, P., Weber, B. (eds.) Advanced Information Systems Engineering - 31st International Conference, CAiSE 2019, Rome, Italy, June 3-7, 2019, Proceedings. Lecture Notes in Computer Science, vol. 11483, pp. 595–610. Springer (2019), `https://doi.org/10.1007/978-3-030-21290-2\_37`
47. Novak, D., Beyeler, B., Omlin, X., Riener, R.: Workload estimation in physical human–robot interaction using physiological measurements. Interacting with computers 27(6), 616–629 (2015)
48. Object Management Group OMG: Case Management Model and Notation Version 1.0 (2014), `http://www.omg.org/spec/CMMN/1.0/PDF`
49. OMG, O.M.G.: Business process modeling notation v 2.0 (2006), `https://www.omg.org/spec/BPMN/2.0/`
50. Paas, F., Tuovinen, J.E., Tabbers, H., Van Gerven, P.W.: Cognitive load measurement as a means to advance cognitive load theory. Educational psychologist 38(1), 63–71 (2003)
51. Paas, F.G., Van Merriënboer, J.J.: Instructional control of cognitive load in the training of complex cognitive tasks. Educational psychology review 6(4), 351–371 (1994)
52. Perugini, D., De Campos, C., Petrelli, M., Morgavi, D., Vetere, F., Dingwell, D.: Quantifying magma mixing with the shannon entropy: Application to simulations and experiments. Lithos 236, 299–310 (2015)
53. Pesic, M., Schonenberg, H., van der Aalst, W.M.: DECLARE: Full Support for Loosely-Structured Processes. In: 11th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007). pp. 287–287. IEEE (oct 2007), `http://ieeexplore.ieee.org/document/4384001/`
54. Petrusel, R., Mendling, J.: Eye-tracking the factors of process model comprehension tasks. In: International Conference on Advanced Information Systems Engineering. pp. 224–239. Springer (2013)
55. Pichler, P., Weber, B., Zugal, S., Pinggera, J., Mendling, J., Reijers, H.A.: Imperative versus declarative process modeling languages: An empirical investigation. In: International Conference on Business Process Management. pp. 383–394. Springer (2011)
56. Pinggera, J., Porcham, T., Zugal, S., Weber, B.: Lipromo-literate process modeling. In: CAiSE Forum. vol. 855, pp. 163–170. Citeseer (2012)
57. Polančič, G., Cegnar, B.: Complexity metrics for process models–a systematic literature review. Computer Standards & Interfaces 51, 104–117 (2017)
58. Reichert, M., Weber, B.: Enabling Flexibility in Process-Aware Information Systems. Springer Berlin Heidelberg, Berlin, Heidelberg (2012), `http://link.springer.com/10.1007/978-3-642-30409-5`
59. Reynoso, L., Rolón, E., Genero, M., García, F., Ruiz, F., Piattini, M.: Formal definition of measures for bpmn models. In: International Workshop on Software Measurement. pp. 285–306. Springer (2009)
60. Sakai, K., Kitaguchi, K., Hikosaka, O.: Chunking during human visuomotor sequence learning. Experimental brain research 152(2), 229–242 (2003)
61. Sánchez-González, L., García, F., Ruiz, F., Mendling, J.: A study of the effectiveness of two threshold definition techniques (2012)
62. Scaife, M., Rogers, Y.: External cognition: how do graphical representations work? International journal of human-computer studies 45(2), 185–213 (1996)
63. Scott, J., Carrington, P.J.: The SAGE handbook of social network analysis. SAGE publications (2011)

64. Shannon, C.E.: A mathematical theory of communication. The Bell system technical journal 27(3), 379–423 (1948)
65. Shao, J., Wang, Y.: A new measure of software complexity based on cognitive weights. Canadian Journal of Electrical and Computer Engineering 28(2), 69–74 (2003)
66. Slaats, T.: Declarative and hybrid process discovery: Recent advances and open challenges. J. Data Semant. 9(1), 3–20 (2020), `https://doi.org/10.1007/s13740-020-00112-9`
67. Slaats, T., Mukkamala, R.R., Hildebrandt, T., Marquard, M.: Exformatics declarative case management workflows as dcr graphs. In: Business process management, pp. 339–354. Springer (2013)
68. Slaats, T., Mukkamala, R.R., Hildebrandt, T., Marquard, M.: Exformatics declarative case management workflows as dcr graphs. In: BPM proceedings (2013)
69. Sternberg, R.J., Sternberg, K.: Cognitive psychology. Nelson Education (2016)
70. Stieff, M., Werner, S., DeSutter, D., Franconeri, S., Hegarty, M.: Visual chunking as a strategy for spatial thinking in stem. Cognitive Research: Principles and Implications 5, 1–15 (2020)
71. Sweller, J.: Cognitive load theory. In: Psychology of learning and motivation, vol. 55, pp. 37–76. Elsevier (2011)
72. Tulving, E., Craik, F.I.: The Oxford handbook of memory. Oxford University Press (2000)
73. Veltman, J., Jansen, C.: The role of operator state assessment in adaptive automation. Tech. rep., TNO DEFENCE SECURITY AND SAFETY SOESTERBERG (NETHERLANDS) (2005)
74. Vinga, S.: Information theory applications for biological sequence analysis. Briefings in bioinformatics 15(3), 376–389 (2014)
75. Vogt, W.P., Johnson, B.: Dictionary of statistics & methodology: A nontechnical guide for the social sciences. Sage (2011)
76. Wohlin, C., Runeson, P., Höst, M., Ohlsson, M.C., Regnell, B., Wesslén, A.: Experimentation in software engineering. Springer Science & Business Media (2012)
77. Yourdon, E., Constantine, L.L.: Structured design: fundamentals of a discipline of computer program and systems design. Prentice-Hall, Inc. (1979)
78. Zhang, D., Ding, Y., Stegall, J., Mo, L.: The effect of visual-chunking-representation accommodation on geometry testing for students with math disabilities. Learning Disabilities Research & Practice 27(4), 167–177 (2012)
79. Zhou, R., Cai, R., Tong, G.: Applications of entropy in finance: A review. Entropy 15(11), 4909–4931 (2013)
80. Zimoch, M., Pryss, R., Schobel, J., Reichert, M.: Eye tracking experiments on process model comprehension: lessons learned. In: Enterprise, Business-Process and Information Systems Modeling, pp. 153–168. Springer (2017)
81. Zugal, S.: Applying Cognitive Psychology for Improving the Creation, Understanding and Maintenance of Business Process Models. Ph.D. thesis, University of Innsbruck (2013)

Bibliography

[ABLL$^+$19] Amine Abbad Andaloussi, Jon Buch-Lorentsen, Hugo A López, Tijs Slaats, and Barbara Weber. Exploring the modeling of declarative processes using a hybrid approach. In *International Conference on Conceptual Modeling*, pages 162–170. Springer, 2019.

[ABS$^+$19] Amine Abbad Andaloussi, Andrea Burattin, Tijs Slaats, Anette Chelina Møller Petersen, Thomas T Hildebrandt, and Barbara Weber. 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.

[ABS$^+$20] Amine Abbad Andaloussi, Andrea Burattin, Tijs Slaats, Ekkart Kindler, and Barbara Weber. On the declarative paradigm in hybrid business process representations: A conceptual framework and a systematic literature study. *Information Systems*, 91:101505, 2020.

[ADB$^+$20] Amine Abbad Andaloussi, Christopher J Davis, Andrea Burattin, Hugo A López, Tijs Slaats, and Barbara Weber. Understanding quality in declarative process modeling through the mental models of experts. In *International Conference on Business Process Management*, pages 417–434. Springer, 2020.

[AG11] Yvonne Lederer Antonucci and Richard J Goeke. Identification of appropriate responsibilities and positions for business process management success. *Business process management Journal*, 2011.

[AKR07] Bjorn Axenath, Ekkart Kindler, and Vladimir Rubin. Amfibia: a meta-model for integrating business process modelling aspects. *International Journal of Business Process Integration and Management*, 2(2):120–131, 2007.

[AS68] Richard C Atkinson and Richard M Shiffrin. Human memory: A proposed system and its control processes. *Psychology of learning and motivation*, 2(4):89–195, 1968.

[ASB$^+$18a] Amine Abbad Andaloussi, Tijs Slaats, Andrea Burattin, T Thomas Hildebrandt, and Barbara Weber. A research model to test the understandability of hybrid process models using dcr graphs. In *EcoKnow Workshop, co-located with the 30th International Conference on Advanced Information Systems Engineering*, 2018. `https://www.forskningsdatabasen.dk/en/catalog/2435479562`.

[ASB$^+$18b] Amine Abbad Andaloussi, Tijs Slaats, Andrea Burattin, Thomas T Hildebrandt, and Barbara Weber. Evaluating the understandability of hybrid process model representations using eye tracking: first insights. In *International Conference on Business Process Management*, pages 475–481. Springer, 2018.

[ASS$^+$20] Amine Abbad Andaloussi, Pnina Soffer, Tijs Slaats, Andrea Burattin, and Barbara Weber. The impact of modularization on the understandability of declarative process models: A research model. In *NeuroIS Retreat*, pages 133–144. Springer, 2020.

[Ast13] Phrasant K Astalin. Qualitative research designs: A conceptual framework. *International journal of social science & interdisciplinary research*, 2(1):118–124, 2013.

[AZB$^+$20] Amine Abbad Andaloussi, Francesca Zerbato, Andrea Burattin, Tijs Slaats, Thomas T Hildebrandt, and Barbara Weber. 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*, pages 1–28, 2020.

[Bad12] Alan Baddeley. Working memory: theories, models, and controversies. *Annual review of psychology*, 63:1–29, 2012.

[BGP06] Zafer Bilda, John S Gero, and Terry Purcell. To sketch or not to sketch? that is the question. *Design studies*, 27(5):587–613, 2006.

[BH74] Alan D Baddeley and Graham Hitch. Working memory. In *Psychology of learning and motivation*, volume 8, pages 47–89. Elsevier, 1974.

[CACS07] Richard A Carlson, Marios N Avraamides, Melanie Cary, and Stephen Strasberg. What do the hands externalize in simple arithmetic? *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 33(4):747, 2007.

[CB07] Kathy Charmaz and Linda Liska Belgrave. Grounded theory. *The Blackwell encyclopedia of sociology*, 2007.

[CFF$^+$18] Flavio Corradini, Alessio Ferrari, Fabrizio Fornari, Stefania Gnesi, Andrea Polini, Barbara Re, and Giorgio O Spagnolo. A guidelines framework for understandable bpmn models. *Data & Know Engineering*, 113:129–154, 2018.

[CK11] Mingyuan Chu and Sotaro Kita. The nature of gestures' beneficial role in spatial problem solving. *Journal of Experimental Psychology: General*, 140(1):102, 2011.

[Cow01] Nelson Cowan. The magical number 4 in short-term memory: A reconsideration of mental storage capacity. *Behavioral and brain sciences*, 24(1):87–114, 2001.

[Cow16] Nelson Cowan. *Working memory capacity: Classic edition.* Psychology press, 2016.

[CSI11] Ran Cheng, Shazia Sadiq, and Marta Indulska. Framework for business process and rule integration: a case of bpmn and sbvr. In *International Conference on Business Information Systems*, pages 13–24. Springer, 2011.

[CZW$^+$16] Fang Chen, Jianlong Zhou, Yang Wang, Kun Yu, Syed Z Arshad, Ahmad Khawaji, and Dan Conway. *Robust multimodal cognitive load measurement.* Springer, 2016.

[DGDMM15] Giuseppe De Giacomo, Marlon Dumas, Fabrizio Maria Maggi, and Marco Montali. Declarative process modeling in bpmn. In *International Conference on Advanced Information Systems Engineering*, pages 84–100. Springer, 2015.

[DH07] Christopher J Davis and Ellen M Hufnagel. Through the eyes of experts: A socio-cognitive perspective on the automation of fingerprint work. *MIS Quarterly*, 31(4), 2007.

[DHMS18] Søren Debois, Thomas Hildebrandt, Morten Marquard, and Tijs Slaats. Hybrid process technologies in the financial sector: the case of brfkredit. In *Business Process Management Cases*, pages 397–412. Springer, 2018.

[DHS18] Søren Debois, Thomas T Hildebrandt, and Tijs Slaats. Replication, refinement & reachability: complexity in dynamic condition-response graphs. *Acta Informatica*, 55(6):489–520, 2018.

[DLRMR13] Marlon Dumas, Marcello La Rosa, Jan Mendling, and Hajo A Reijers. *Business process management.* Springer, 2013.

[DLS$^+$20] Søren Debois, Hugo A López, Tijs Slaats, Amine Abbad Andaloussi, and Thomas T Hildebrandt. Chain of events: Modular process models for the law. In *International Conference on Integrated Formal Methods*, pages 368–386. Springer, 2020.

[DN70] K Denzin Norman. The research act in sociology. *USA, Illinois*, 1970.

[DSDWSV16] Johannes De Smedt, Jochen De Weerdt, Estefanía Serral, and Jan Vanthienen. Improving understandability of declarative process models by revealing hidden dependencies. In *International Conference on Advanced Information Systems Engineering*, pages 83–98. Springer, 2016.

[DSDWSV18] Johannes De Smedt, Jochen De Weerdt, Estefanía Serral, and Jan Vanthienen. Discovering hidden dependencies in constraint-based declarative process models for improving understandability. *Information Systems*, 74:40–52, 2018.

[DSDWVP16] Johannes De Smedt, Jochen De Weerdt, Jan Vanthienen, and Geert Poels. Mixed-paradigm process modeling with intertwined state spaces. *Business & Information Systems Engineering*, 58(1):19–29, 2016.

[DV11] Frank Dengler and Denny Vrandečić. Wiki-based maturing of process descriptions. In *International Conference on Business Process Management*, pages 313–328. Springer, 2011.

[DVDA04] Juliane Dehnert and Wil MP Van Der Aalst. Bridging the gap between business models and workflow specifications. *International Journal of Cooperative Information Systems*, 13(03):289–332, 2004.

[ETvdHP16] Amal Elgammal, Oktay Turetken, Willem-Jan van den Heuvel, and Mike Papazoglou. Formalizing and appling compliance patterns for business process compliance. *Software & Systems Modeling*, 15(1):119–146, 2016.

[Fig17] Kathrin Figl. Comprehension of procedural visual business process models. *Business & Information Systems Engineering*, 59(1):41–67, 2017.

[FLM$^+$09] Dirk Fahland, Daniel Lübke, Jan Mendling, Hajo Reijers, Barbara Weber, Matthias Weidlich, and Stefan Zugal. Declarative versus imperative process modeling languages: The issue of understandability. In *Enterprise, Business-Process and Information Systems Modeling*, pages 353–366. Springer, 2009.

[GBS07] Tobias Graml, Ralf Bracht, and Marcus Spies. Patterns of business rules to enable agile business processes. In *11th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007)*, pages 365–365. IEEE, 2007.

[Gil15] Sam J Gilbert. Strategic use of reminders: Influence of both domain-general and task-specific metacognitive confidence, independent of objective memory ability. *Consciousness and Cognition*, 33:245–260, 2015.

[GMNKW01] Susan Goldin-Meadow, Howard Nusbaum, Spencer D Kelly, and Susan Wagner. Explaining math: Gesturing lightens the load. *Psychological science*, 12(6):516–522, 2001.

[Gra06] P.O. Gray. *Psychology.* Worth Publishers, 2006.

[Gui05] Giancarlo Guizzardi. Ontological foundations for structural conceptual models. 2005.

[HAC$^+$20] Thomas T Hildebrandt, Amine Abbad Andaloussi, Lars R Christensen, Søren Debois, Nicklas Pape Healy, Hugo A López, Morten Marquard, Naja LH Møller, Anette CM Petersen, Tijs Slaats, et al. Ecoknow: Engineering effective, co-created and compliant adaptive case management systems for knowledge workers. In *Proceedings of the International Conference on Software and System Processes*, pages 155–164, 2020.

[Hin16] Knut Hinkelmann. Business process flexibility and decision-aware modeling—the knowledge work designer. In *Domain-specific conceptual modeling*, pages 397–414. Springer, 2016.

[HM11] Thomas T Hildebrandt and Raghava Rao Mukkamala. Declarative event-based workflow as distributed dynamic condition response graphs. *arXiv preprint arXiv:1110.4161*, 2011.

[HMS11] Thomas Hildebrandt, Raghava Rao Mukkamala, and Tijs Slaats. Nested dynamic condition response graphs. In *International conference on fundamentals of software engineering*, pages 343–350. Springer, 2011.

[HNA$^+$11] Kenneth Holmqvist, Marcus Nyström, Richard Andersson, Richard Dewhurst, Halszka Jarodzka, and Joost Van de Weijer. *Eye tracking: A comprehensive guide to methods and measures*. OUP Oxford, 2011.

[HSFM06] Janelle B Hill, Jim Sinur, David Flint, and Michael James Melenovsky. Gartner's position on business process management. *Gartner Research G*, 136533, 2006.

[HSLW18] Cornelia Haisjackl, Pnina Soffer, Shao Yi Lim, and Barbara Weber. How do humans inspect bpmn models: an exploratory study. *Software & Systems Modeling*, 17(2):655–673, 2018.

[JC80] Marcel A Just and Patricia A Carpenter. A theory of reading: From eye fixations to comprehension. *Psychological review*, 87(4):329, 1980.

[Kal09] Slava Kalyuga. The expertise reversal effect. In *Managing cognitive load in adaptive multimedia learning*, pages 58–80. IGI Global, 2009.

[KC07] Barbara Kitchenham and Stuart Charters. Guidelines for performing systematic literature reviews in software engineering. 2007.

[Kel55] George A Kelly. The psychology of personal constructs norton. *New York*, 1955.

[KKN12] Krzysztof Kluza, Krzysztof Kaczor, and Grzegorz J Nalepa. Enriching business processes with rules using the oryx bpmn editor. In *International Conference on Artificial Intelligence and Soft Computing*, pages 573–581. Springer, 2012.

[KN19] Krzysztof Kluza and Grzegorz J Nalepa. Formal model of business processes integrated with business rules. *Information Systems Frontiers*, 21(5):1167–1185, 2019.

[KNS92] G Keller, M Nüttgens, and A W Scheer. *Semantische Prozessmodellierung auf der Grundlage "ereignisgesteuerter Prozessketten (EPK)"*. Institut für Wirtschaftsinformatik Saarbrücken: Veröffentlichungen des Instituts für Wirtschaftsinformatik. Inst. für Wirtschaftsinformatik, 1992.

[Kro16] John Krogstie. *Quality in Business Process Modeling*. Springer, Cham, 2016.

[LDHM18] Hugo A López, Søren Debois, Thomas T Hildebrandt, and Morten Marquard. The process highlighter: From texts to declarative processes and back. *BPM (Dissertation/Demos/Industry)*, 2196:66–70, 2018.

[LDSH20] Hugo A López, Søren Debois, Tijs Slaats, and Thomas T Hildebrandt. Business process compliance using reference models of law. In *FASE*, pages 378–399, 2020.

[Llo94] John W Lloyd. Practical advtanages of declarative programming. In *GULP-PRODE (1)*, pages 18–30, 1994.

[LMMS19] Hugo A López, Morten Marquard, Lukas Muttenthaler, and Rasmus Strømsted. Assisted declarative process creation from natural language descriptions. In *2019 IEEE 23rd International Enterprise Distributed Object Computing Workshop (EDOCW)*, pages 96–99. IEEE, 2019.

[Man17] Object Management Group. An OMG ® Semantics of Business Vocabulary and Business Rules TM Publication-Content-Model-for-SBVR.xml. 2017.

[MB13] Eglė Mickevičiūtė and Rimantas Butleris. Towards the combination of bpmn process models with sbvr business vocabularies and rules. In *International Conference on Information and Software Technologies*, pages 114–121. Springer, 2013.

[Men07] Jan Mendling. *Detection and prediction of errors in EPC business process models*. PhD thesis, Wirtschaftsuniversität Wien Vienna, 2007.

[Men09] Jan Mendling. Empirical studies in process model verification. In *Transactions on petri nets and other models of concurrency II*, pages 208–224. Springer, 2009.

[MGR11] Milan Milanovic, Dragan Gasevic, and Luis Rocha. Modeling flexible business processes with business rule patterns. In *2011 IEEE 15th International Enterprise Distributed Object Computing Conference*, pages 65–74. IEEE, 2011.

[Mil56] George A Miller. The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81, 1956.

[ML00] David Marsden and Dale Littler. Exploring consumer product construct systems with the repertory grid technique. *Qualitative Market Research*, 2000.

[Moo09] Daniel Moody. The “physics” of notations: toward a scientific basis for constructing visual notations in software engineering. *IEEE Transactions on SE*, 2009.

[MRvdA10] Jan Mendling, Hajo A Reijers, and Wil MP van der Aalst. Seven process modeling guidelines (7pmg). *Information and Software Technology*, 52(2):127–136, 2010.

[MS93] JH McMillan and Sally Schumacher. Research in education: A conceptual understanding. *New York: Haprer Collins*, 1993.

[MS02] Peter Mangan and Shazia Sadiq. On building workflow models for flexible processes. In *ACM International Conference Proceeding Series*, volume 18, pages 103–109. Citeseer, 2002.

[MSS16] Morten Marquard, Muhammad Shahzad, and Tijs Slaats. Web-based modelling and collaborative simulation of declarative processes. In *International Conference on Business Process Management*, pages 209–225. Springer, 2016.

[Muk12] Raghava Rao Mukkamala. A formal model for declarative workflowsdynamic condition response graphs. 2012.

[Nor18] Håkon Normann. *Independence, Name-passing and Constraints in Models for Concurrency.* IT University of Copenhagen, Computer Science Department, 2018.

[Obe09] Klaus Oberauer. Design for a working memory. *Psychology of learning and motivation*, 51:45–100, 2009.

[Obj16] Object Management Group CMMN Technical Committee. Case Management Model and Notation, version 1.1, 2016. `https://www.omg.org/spec/CMMN/1.1/PDF`.

[OMG06] Object Management Group OMG. Business process modeling notation v 2.0. 2006.

[Pai91] Allan Paivio. Dual coding theory: Retrospect and current status. *Canadian Journal of Psychology/Revue canadienne de psychologie*, 45(3):255, 1991.

[PC17] Gregor Polančič and Blaž Cegnar. Complexity metrics for process models–a systematic literature review. *Computer Standards & Interfaces*, 51:104–117, 2017.

[Pet62] Carl Adam Petri. Kommunikation mit Automaten, 1962.

[Pin14] Jakob Pinggera. *The Process of Process Modeling.* PhD thesis, Department of Computer Science, 2014.

[PPZW12] Jakob Pinggera, Thomas Porcham, Stefan Zugal, and Barbara Weber. Lipromo-literate process modeling. In *CAiSE Forum*, volume 855, pages 163–170. Citeseer, 2012.

[PRS03] Fred Paas, Alexander Renkl, and John Sweller. Cognitive load theory and instructional design: Recent developments. *Educational psychologist*, 38(1):1–4, 2003.

[PSVdA07] Maja Pesic, Helen Schonenberg, and Wil MP Van der Aalst. Declare: Full support for loosely-structured processes. In *11th IEEE International Enterprise Distributed Object Computing Conference (EDOC 2007)*, pages 287–287. IEEE, 2007.

[PTTVG03] Fred Paas, Juhani E Tuovinen, Huib Tabbers, and Pascal WM Van Gerven. Cognitive load measurement as a means to advance cognitive load theory. *Educational psychologist*, 38(1):63–71, 2003.

[PVM94] Fred GWC Paas and Jeroen JG Van Merriënboer. Instructional control of cognitive load in the training of complex cognitive tasks. *Educational psychology review*, 6(4):351–371, 1994.

[PZW$^+$11] Jakob Pinggera, Stefan Zugal, Matthias Weidlich, Dirk Fahland, Barbara Weber, Jan Mendling, and Hajo A Reijers. Tracing the process of process modeling with modeling phase diagrams. In *International Conference on Business Process Management*, pages 370–382. Springer, 2011.

[RG16] Evan F Risko and Sam J Gilbert. Cognitive offloading. *Trends in cognitive sciences*, 20(9):676–688, 2016.

[Rit07] Peter Rittgen. Negotiating models. In *International Conference on Advanced Information Systems Engineering*, pages 561–573. Springer, 2007.

[RMCK14] Evan F Risko, Srdan Medimorec, Joseph Chisholm, and Alan Kingstone. Rotating with rotated text: A natural behavior approach to investigating cognitive offloading. *Cognitive Science*, 38(3):537–564, 2014.

[RMR10] Hajo A Reijers, Jan Mendling, and Jan Recker. Business process quality management. In *Handbook on Business Process Management 1*, pages 167–185. Springer, 2010.

[RRIG09] Jan Recker, Michael Rosemann, Marta Indulska, and Peter Green. Business process modeling-a comparative analysis. *Journal of the association for information systems*, 10(4):1, 2009.

[Rum17] David E Rumelhart. Schemata: The building blocks. *Theoretical issues in reading comprehension: Perspectives from cognitive psychology, linguistics, artificial intelligence and education*, 11(1):33–58, 2017.

[RW12] Manfred Reichert and Barbara Weber. *Enabling flexibility in process-aware information systems: challenges, methods, technologies.* Springer Science & Business Media, 2012.

[Sal15] Johnny Saldaña. *The coding manual for qualitative researchers.* Sage, 2015.

[SC94] John Sweller and Paul Chandler. Why some material is difficult to learn. *Cognition and instruction*, 12(3):185–233, 1994.

[SFDA$^+$20] Josep Sanchez-Ferreres, Luis Delicado, Amine Abbab Andaloussi, Andrea Burattin, Guillermo Calderon-Ruiz, Barbara Weber, Josep Carmona, and Lluís Padró. Supporting the process of learning and teaching process models. *IEEE Transactions on Learning Technologies*, 2020.

[SHCV15] Tijs Slaats, Thomas T Hildebrandt, Marco Carbone, and Hagen Völzer. *Flexible process notations for cross-organizational case management systems.* PhD thesis, 2015.

[SKW11] Pnina Soffer, Maya Kaner, and Yair Wand. Towards understanding the process of process modeling: theoretical and empirical considerations. In *International Conference on Business Process Management*, pages 357–369. Springer, 2011.

[SMHM13] Tijs Slaats, Raghava Rao Mukkamala, Thomas Hildebrandt, and Morten Marquard. Exformatics declarative case management workflows as dcr graphs. In *Business process management*, pages 339–354. Springer, 2013.

[SR96] Mike Scaife and Yvonne Rogers. External cognition: how do graphical representations work? *International journal of human-computer studies*, 45(2):185–213, 1996.

[SR05] Mike Scaife and Yvonne Rogers. External cognition, innovative technologies, and effective learning. *Cognition, education and communication technology*, pages 181–202, 2005.

[SS16] Robert J Sternberg and Karin Sternberg. *Cognitive psychology*. Nelson Education, 2016.

[SSF81] Valerie Stewart, Andrew Stewart, and Nickie Fonda. *Business applications of repertory grid*. McGraw-Hill London, 1981.

[SSMR16] Tijs Slaats, Dennis MM Schunselaar, Fabrizio M Maggi, and Hajo A Reijers. The semantics of hybrid process models. In *OTM Confederated International Conferences" On the Move to Meaningful Internet Systems"*, pages 531–551. Springer, 2016.

[SSO01] Shazia Sadiq, Wasim Sadiq, and Maria Orlowska. Pockets of flexibility in workflow specification. In *International Conference on Conceptual Modeling*, pages 513–526. Springer, 2001.

[SVMP98] John Sweller, Jeroen JG Van Merrienboer, and Fred GWC Paas. Cognitive architecture and instructional design. *Educational psychology review*, 10(3):251–296, 1998.

[SvS10] Brahmananda Sapkota and Marten van Sinderen. Exploiting rules and processes for increasing flexibility in service composition. In *2010 14th IEEE International Enterprise Distributed Object Computing Conference Workshops*, pages 177–185. IEEE, 2010.

[Swe88] John Sweller. Cognitive load during problem solving: Effects on learning. *Cognitive science*, 12(2):257–285, 1988.

[Swe11] John Sweller. Cognitive load theory. In *Psychology of learning and motivation*, volume 55, pages 37–76. Elsevier, 2011.

[TH02] Felix B Tan and M Gordon Hunter. The repertory grid technique: A method for the study of cognition in information systems. *MIS quarterly*, pages 39–57, 2002.

[The15] The Danish Ministry of Social Affairs and the Interior. Consolidation Act on Social Services, September 2015. Executive Order no. 1053 of 8 September 2015; File no. 2015-4958.

[TS09] Barbara Tversky and Masaki Suwa. Thinking with sketches. 2009.

[Tve02] Barbara Tversky. What do sketches say about thinking. In *2002 AAAI Spring Symposium, Sketch Understanding Workshop, Stanford University, AAAI Technical Report SS-02-08*, pages 148–151, 2002.

[UE07] Nash Unsworth and Randall W Engle. The nature of individual differences in working memory capacity: active maintenance in primary memory and controlled search from secondary memory. *Psychological review*, 114(1):104, 2007.

[VDA16] Wil Van Der Aalst. *Process mining: Data science in action.* Springer, 2016.

[vdAP06] Wil M. P. van der Aalst and Maja Pesic. DecSerFlow: Towards a truly declarative service flow language. In *WS-FM 2006*, volume 4184 of *LNCS*, pages 1–23, Vienna, Austria, 2006. Springer.

[VEIP08] Tim Van Eijndhoven, Maria-Eugenia Iacob, and María Laura Ponisio. Achieving business process flexibility with business rules. In *2008 12th International IEEE Enterprise Distributed Object Computing Conference*, pages 95–104. IEEE, 2008.

[Ves91] Iris Vessey. Cognitive fit: A theory-based analysis of the graphs versus tables literature. *Decision sciences*, 22(2):219–240, 1991.

[VRH+04] Peter Van-Roy, Seif Haridi, et al. *Concepts, techniques, and models of computer programming.* MIT press, 2004.

[Wes19] Mathias Weske. *Business Process Management: Concepts, Languages, Architectures.* Springer, 2019.

[Win75] Terry Winograd. Frame representations and the declarative/procedural controversy. In *Representation and understanding*, pages 185–210. Elsevier, 1975.

[WISW17] Wei Wang, Marta Indulska, Shazia Sadiq, and Barbara Weber. Effect of linked rules on business process model understanding. In *International conference on business process management*, pages 200–215. Springer, 2017.

[WN65] Nancy C Waugh and Donald A Norman. Primary memory. *Psychological review*, 72(2):89, 1965.

[WRH+12] Claes Wohlin, Per Runeson, Martin Höst, Magnus C Ohlsson, Björn Regnell, and Anders Wesslén. *Experimentation in software engineering.* Springer Science & Business Media, 2012.

[WRMR11] Barbara Weber, Manfred Reichert, Jan Mendling, and Hajo A Reijers. Refactoring large process model repositories. *Computers in Industry*, 62(5):467–486, 2011.

[WS13] Michael Westergaard and Tijs Slaats. Mixing paradigms for more comprehensible models. In *Business Process Management*, pages 283–290. Springer, 2013.

[Zha97] Jiajie Zhang. The nature of external representations in problem solving. *Cognitive science*, 21(2):179–217, 1997.

[Zhe17] Robert Z Zheng. *Cognitive load measurement and application: a theoretical framework for meaningful research and practice*. Routledge, 2017.

[ZHPW13] Stefan Zugal, Cornelia Haisjackl, Jakob Pinggera, and Barbara Weber. Empirical evaluation of test driven modeling. *International Journal of Information System Modeling and Design (IJISMD)*, 4(2):23–43, 2013.

[ZPW11] Stefan Zugal, Jakob Pinggera, and Barbara Weber. Creating declarative process models using test driven modeling suite. In *International Conference on Advanced Information Systems Engineering*, pages 16–32. Springer, 2011.

[ZPW12] Stefan Zugal, Jakob Pinggera, and Barbara Weber. Toward enhanced life-cycle support for declarative processes. *Journal of Software: Evolution and Process*, 24(3):285–302, 2012.

[Zug13] Stefan Zugal. *Applying Cognitive Psychology for Improving the Creation, Understanding and Maintenance of Business Process Models*. PhD thesis, University of Innsbruck, 2013.