What we know and what we do not know about DMN

Kathrin Figl\textsuperscript{a}, Jan Mendling\textsuperscript{a,b}, Gul Tokdemir\textsuperscript{c}, Jan Vanthienen\textsuperscript{d}

\textsuperscript{a} University of Innsbruck, Austria
\textsuperscript{b} Wirtschaftsuniversität Wien, Vienna, Austria
\textsuperscript{c} Cankaya University, Ankara, Turkey
\textsuperscript{d} KU Leuven, Belgium

Abstract. The recent Decision Model and Notation (DMN) establishes business decisions as first-class citizens of executable business processes. This research note has two objectives: first, to describe DMN’s technical and theoretical foundations; second, to identify research directions for investigating DMN’s potential benefits on a technological, individual and organizational level. To this end, we integrate perspectives from management science, cognitive theory and information systems research.

Keywords. DMN • BPMN • Process Modeling

Communicated by A. Koschmider. Received 2017-06-18. Accepted after 2 revisions on 2018-03-14.

1 Introduction

The Decision Model and Notation (DMN) is a recent standard of the Object Management Group (2016). It complements the Business Process Model and Notation (BPMN) with a notation for modeling decision logic and dependencies between decisions and data elements. The specification formulates several goals, which can also be understood as hypothetical benefits: First, the notation should be readily understandable by both business users and technical developers. Second, it should be straight forward to transform it to artifacts that implement decision logic. Third, it should be easily usable together with BPMN. DMN enjoys an increasing uptake in industry and receives attention in academic research. However, empirical research on DMN is still scarce such that it is unclear to which degree the proclaimed benefits materialize.

The aim of this paper is to structure future research on DMN. Sect. 2 summarizes the background of DMN. Sect. 3 describes a research agenda for DMN, before Sect. 4 concludes the paper.

2 Background of DMN

DMN is a standard for representing operational decisions of day-to-day business operations. Such decisions are frequently taken and repetitive in nature, e.g., determining if a customer is eligible for an insurance cover. Common operational decisions often relate to the calculation and evaluation of business opportunities, risk management and fraud detection. DMN complements BPMN, which does not model the decision logic in detail. DMN decouples decisions and control flow logic and it opens room for dynamic management of decisions. In most of the process models, decisions are embedded within the models and scattered over process model constructs, eventually posing difficulty in maintainability (Janssens et al. 2016b). In this sense, DMN reduces complexity and provides a decision model which is more precise and clear (Bossuyt and Gailly 2017). In this way, DMN helps business users in controlling their processes and organizational decisions more efficiently and effectively by means of well-designed decision and information structures.

More specifically, DMN defines three aspects of decisions: the decision requirements level, the decision logic level, and the expression language.
Fig. 1 illustrates these levels by the help of an example on a seller’s credit warranting process for a potential buyer. DMN can be used together with BPMN as shown in the figure or independent of business processes.

First, the Decision Requirements Diagram (DRD) represents the relationship between decisions through their information requirements and defines decision requirements through constructs of Decision, Business Knowledge Model, Knowledge Source, Input Data, Information Requirement, Knowledge Requirement and Authority requirement (Object Management Group 2016). In Fig. 1, the input data for decision making is obtained from two sources: namely Findex and Project Specs. There is one decision whose result is used in the business process Credit Sales and two intermediate decisions Credit Eligibility and Cheque Performance, which yield results as input for the final Credit Sales decision. Payable Cheque Criteria is based on due cheque payments of the buyer within one year with respect to project value and project payment term. Paid Cheque Criteria on the other hand depends on previous cheque payment history with respect to project value.

Second, the Decision Logic Level (DLL) represents the logic of a single decision in the form of a boxed expression. One of the most widely used representations for decision logic is a decision table, but other expressions are allowed, e.g. using analytical models, mathematical functions or decision rules. Decision tables define the production rules from input parameters to the output parameters. In Fig. 1, the decision logic for Credit Eligibility is shown as a decision table, in which the parameters Project Value, Total Credibility Amount, Available Credit Limit, Bank Credit Warrant Letter are used as input for determining the output parameter Credit Eligibility.

Third, DMN also standardizes the expression language FEEL (Friendly Enough Expression Language) and a simple subset S-FEEL for use in decision tables. FEEL defines a syntax for expressions, which permits the description of decision logic in terms of decision tables, analytical models, or business rules. At the bottom of Fig. 1, the decision logic of Credit Eligibility is shown using FEEL syntax for the rules expressed in the table.

3 Research Agenda for Investigating DMN Benefits

In this section, we review the literature and discuss a research agenda for investigating the potential benefits of DMN. We structure this discussion by the help of the information systems research framework by Hevner et al. (2004), which identifies technology (Sect. 3.1), individual (Sect. 3.2) and organization (Sect. 3.3) as three relevant areas.

3.1 Research Directions on Technological Benefits of DMN

The history of operational decision management and DMN finds its origin in decision table modeling, where rules for decision logic are represented in a structure of related tables, which map combinations of inputs to outcomes. Decision tables and the accompanying methodology have proven a powerful vehicle for acquiring the decision knowledge and for checking completeness, correctness and consistency (CODASYL Decision Table Task Group 1982). DMN builds upon these concepts and goes further by standardizing existing decision table formats (using a hit policy indicator), by elaborating the requirements diagram, and by introducing a standard expression language. Even though DMN standardizes and extends the modeling capabilities of decision requirements and decision logic (e.g. by adding FEEL), various results from previous research into decision tables can be readily adopted.

Verification & Validation (V&V): Verification and validation of rule-based systems (including decision tables) has been a major area of research, as exemplified by the earlier EUROVAV series of conferences (European Conference on Verification & Validation of Knowledge-based systems) (Antoniou et al. 1998). This is important because at the decision logic level, decision logic is often expressed in rules and tables. There are numerous works dealing with V&V of a set of rules (as present in
single decision tables). Typical rule anomalies are: redundancy (including duplicates and subsumption), ambivalence, circularity, and deficiency (missing rules). Numerous algorithms are available for checking and eliminating contradictions, redundancies and missing rules for all possible values of the input variables. Other approaches have been designed to strictly avoid table anomalies by recommending unique single hit tables. Thus, there has been an increased interest and work in relation to verification and validation studies in literature within the last few decades. For example, various tools have already included anomaly detection algorithms (Hinkelmann 2016). In a similar manner, Calvanese et al. (2016) and Laurson and Maggi (2017) propose new algorithms to discover overlapping and missing rules during DMN table modeling task.

In terms of correctness of decision logic, Batoulis et al. (2017) analyze process models with DMN by checking different soundness levels for decision-aware processes. For an overview of earlier research in this area, see Vanthienen et al. (1998), or more recently Calvanese et al. (2016).

V&V over table networks: Also, V&V of table structures has been covered in earlier research. When input conditions or outcomes are repeated in more than one decision, some parts of the decision logic in a certain decision may become unreachable or inconsistent for specific input values. Checking consistency and completeness between interconnected decision tables, i.e. over rule chains, is a much more challenging problem than verification of single tables.
See Vanthienen et al. (1997) for an overview of inter-tabular verification.

**Table Simplification:** Decision tables can be simplified and split up using normalization (Vanthienen and Dries 1994). The first type of simplification means that rules with equal condition inputs for all but one condition and with the same outcomes can be joined together, reducing the number of rules in the table. This is called table contraction, minimizing the number of rows for the given condition order. For this case, a recent study by Laurson and Maggi (2017) is a good example, where a rule merging algorithm is proposed for table simplification. One can also identify the order of the input conditions that leads to the most compact table, thereby optimizing the condition order. Another simplification is to split complex tables into more simple ones. Decision tables can (or should) be split up if the outcomes are not dependent on all the conditions. This is called factoring or normalization, analogous to normalization in relational database theory where attributes should be dependent on the key.

**Code Generation:** When properly specified using (S)FEEL, decision models and tables are executable, given that appropriate DMN tooling is available. This is a straightforward execution without further optimization. In a number of cases, however, attention could be paid to execution efficiency or more flexible forms of code generation. Since the early days of decision tables, a lot of work has been done in this area, by transforming decision tables into optimal code, by generating least-cost execution trees based on condition test times and case frequencies, see e.g. Lew (1978) or CODASYL Decision Table Task Group (1982).

**Decision Mining:** Operational decisions can be modeled in DMN by domain experts by using the domain knowledge present in e.g. rules, procedures, policies and regulations. But decision logic can also be derived from case data where the mined model is discovered or transformed into a decision table (Baesens et al. 2003; Wets et al. 1998). Not unlike process models, which can be discovered from events logs, decision discovery is a form of knowledge discovery from logs containing historical data about case attributes and their outcome. Currently, decision mining is often limited to discovering the decision logic at a certain decision point in a process model, but a more complex challenge is the integrated mining of both a process and a decision model based on extensive decision process logs (Smedt et al. 2017a).

Accordingly, literature has provided numerous research on how to create an integrated model with decisions and processes. For example, Biard et al. (2015), in their study recommend defining a decision task for multiple gateways in the process model and constructing a decision model as a separated entity. They emphasize that DMN’s scope is restricted to operational level decisions, instead of tactical ones, as they are related to pre-defined decisions. In the recent research, Bazhenova (2017) describes how to extract a decision model from process model based on split gateways and event logs. In another study, Batoulis et al. (2016) bring forward an approach to adjust decision logic dynamically, using event log information during process implementation by creating DMN model automatically that will improve process execution consequently. From an alternative perspective, van der Aa et al. (2016) create BPMN/DMN models based on data-flow structure automatically. Similarly, an approach for automatic DMN construction is defined by Bazhenova and Weske (2015), where decision logic has been extracted from event logs of a simple process model in banking domain. The research extracts decisions from process models based on local decision points and limited in terms of applying their proposal to simple process models in a specific domain. Mertens et al. (2017) introduce DeciClare, which is a mixed-perspective process modeling language. It is more for loosely framed processes which incorporates functional, control-flow, data and resource views and includes concepts of DMN.
language for modeling data perspective. Hence, it provides a comprehensive view for integration of decisions and processes. In regard to composition and decomposition of processes and decisions, recent research mostly focuses on simpler process models, where decisions are local, and do not span across process elements other than gateways. However, complex process models, where decisions may extend over process modeling elements and where dependencies exist, a holistic approach is required for integrated modeling of processes and decisions (Hasic et al. 2018). While in literature the number of such research remains low, recent studies acknowledge this requirement and propose enhanced methods. For example, Smedt et al. (2017b) put forward a holistic approach for decision extraction from process models called Process Mining Integrating Decisions (P-MInD), which incorporates various business activities affecting the decision perspective.

DMN promises various benefits for the efficient and effective design and management of decisions, e.g. in business processes. As recent research suggests, decision modeling alongside business processes enables and helps business users to manage complexity (Janssens et al. 2016a; Taylor 2011). This better management of complexity should also help to support flexibility and maintainability of processes.

Research challenges arise in this context regarding the consistency between DMN and BPMN. How should decisions and processes be modeled in an integrated way? How can decisions and processes be mined together in a combined way to reveal entire decision structures? How can we transform decision logic from BPMN models to DMN? Answering these questions requires research methods that are grounded in formal science and design science.

3.2 Research Directions on Individual Benefits of DMN

In order to structure the discussion of DMN-related research problems on the individual level, we refer to a theoretical model by Gemino and Wand (2003) that describes modeling as a process of knowledge construction. The outcome of this process is influenced by three major perspectives: First, the characteristics of model viewers in association with their tasks (Sect. 3.2.1); second, the content that is captured in the model (Sect. 3.2.2) and third, the presentation format of this content (Sect. 3.2.3). From a cognitive point of view, the content view relates to the inherent complexity of information that must be understood (Sweller 2010). While intrinsic cognitive load may not be easily altered without changing the decision situation, extraneous cognitive load can be decreased by how the decision model is presented and more cognitive effort can be devoted to schema construction (germane load) (Sweller 2010).

3.2.1 Characteristics of Model Viewers

The DMN specification (Object Management Group 2016, p. 169) lists three types of possible user groups: business analysts, business users and technical developers. These user groups have different technical expertise and they focus either on creating or on reading DMN models, respectively.

Novice versus Expert: Prior research on conceptual modeling has investigated expertise from different angles. Studies including Schenk et al. (1998) have observed striking differences in the way how novice system analysts approach a project (rather bottom-up and opportunity-driven) as compared to expert system analysts (rather top-down and goal-oriented). There are various requirements for a person to transition from a novice to an expert status, likely also for DMN: learning the language and its concepts, developing patterns of how to capture recurring problems, and deliberate training over a longer period of time. The roles mentioned in Object Management Group (2016, p. 169), i.e. business analysts designing decisions, business users populating decision models and technical developers mapping business terms to appropriate data technologies have different skills and prior knowledge with respect to decision modeling. Also, the distinction of S-FEEL
for business users and full FEEL for advanced
business analysts or developers emphasizes the
different skill sets. Concerning domain expertise,
decision making requires high levels of
domain expertise (Bock 2015). Thus, in gen-
eral, decisions modeled in DMN also have the
advantage to externalize such knowledge for
employees less familiar with a decision domain.

Reading and Creating Models: More generally, we know from research on expertise that
being an expert is very much bound to a specific
task (Ericsson and Lehmann 1996). Conceptual modeling research has distinguished the
activities of creating and reading a model. In
essence, a model viewer has to be familiar with
the syntax and semantics of a notation like
DMN in order to interpret individual models.
The task of modelers is more challenging, since
they have to transform ideas, observations
and discussions into a correct representation.
Tasks of verification and validation are highly
important in this context, and different types of
users might be unequally skilled for conducting
them.

Besides foreseen advantages, the understand-
ability of DMN by different stakeholders de-
serves profound attention. In this sense, adapt-
ing DMN into organizational decision making
processes necessitates stakeholders like busi-
ness analysts or IT professionals to get famil-
lar with the notation. Research could suggest
modifications to DMN to improve the notation
and to shorten the learning curve of stakeholders in trainings. Questions arise here on how
and in which circumstances users can most
effectively work with DMN and which char-
acteristics of users and models best facilitate
understanding. Answering these questions re-
quires research methods that are grounded in
empirical research.

3.2.2 Semantic Content of DMN
The semantic constructs of DMN are defined by
the metamodel and have already been discussed
in the previous sections. Now, we want to out-
line some directions in which future research on
the cognitive difficulty of different semantic con-
structs could be pursued. Similar research has
been conducted e.g. in the area of cognitive dif-
ficulty of control flow patterns used in process
models (Figl and Laue 2015) or of different types
of features and relations used in variability models
(Reinhartz-Berger et al. 2014). If future research
is able to determine valid and reliable values for
the cognitive difficulty of understanding specific
parts of decisions, such values could then be used
to guide modeling tool developers to provide feed-
back on the cognitive difficulty of models to users.

Model editors could calculate global metrics for
the complexity of a decision, warn the modeler
when they exceed a certain threshold and use color
highlighting of models and decision tables to vi-
sually highlight difficult parts. Since DRGs do
not depict detail information on how the decisions
are exactly taken, we deem the investigation of
the cognitive difficulty of decision tables more
promising. For instance, future work could em-
pirically assess whether unique/first hit/any hit
or priority hit policies are more complex to un-
derstand and lead to higher error rates. While
automated algorithms might check rule anom-
alies, still human comprehension of the rules is
necessary for a variety of tasks.

3.2.3 Visual Presentation of DMN
When considering the visual presentation of DMN,
we have to look separately at the three levels
(DRD, DLL and FEEL). Although these levels are
related to each other, their representation format–
graphical models, tables and textual expression
language–varies significantly. In the context of
this paper, we focus on DRDs and decision ta-
bles, but do not discuss the FEEL, because it is
mainly textual. We structure our discussion of the
presentation of DMN into various sections based
on the physics of notations framework (Moody
2009), which integrates different theoretical per-
spectives to define nine principles how to design
visual notations that do not cause more cognitive
load for users than necessary. These principles
are semiotic clarity, graphic economy, visual expressiveness and perceptual discriminability, semantic transparency, dual coding, cognitive fit, complexity management and cognitive integration. Furthermore, we add one aspect which is not explicitly defined at the notational level, but plays a critical role at the level of single diagrams: labeling and naming conventions (Leopold et al. 2013). We consider primary notation which would refer to the standard document published by OMG and relevant aspects of secondary notation, which relates to “things which are not formally part of a notation which are nevertheless used to interpret it” (Petre 2006, p. 293). Dangarska et al. (2016) have presented the first analysis of DRGs (decision tables were not evaluated) according to Moody’s framework based on an expert assessment. Besides expert evaluations, future research could e.g. conduct questionnaire-based studies to provide user evaluations of the symbol set of DMN, e.g. by using scales to assess semiotic clarity, perceptual discriminability, visual expressiveness or semantic transparency (see for instance evaluations of symbol sets of other modeling notations (Figl et al. 2010, 2013)). Another approach could be to develop symbol sets optimized based on cognitive design guidelines, which has been done for other existing notations (Genon et al. 2012, 2011) and compare their effect on comprehensibility with original DMN symbols. Moreover, usability tests and experiments including eye-tracking could be performed by researchers to assess the understandability of the visual notation of DMN. The following sections gives an background on relevant factors which could be included in experiments and highlights relevant open research questions.

Semiotic Clarity

The principle of semiotic clarity demands that there is a 1:1 relationship between any semantic construct and the corresponding visual symbol the notation offers. One potential violation of this rule in the form of symbol redundancy (more than one visual representation for one and the same underlying semantic construct) can for instance be found in the DMN notation (Object Management Group 2016, p. 31): “An alternative compliant way to display requirements for Input Data, especially useful when DRDs are large or complex, is that Input Data are not drawn as separate notational elements in the DRD, but are instead listed on those Decision elements which require them.” In the context of semantic constructs, a representational analysis according to the theory by Wand/Weber based on Bunge (Recker et al. 2011) may also be effective to highlight the concepts which DMN is supporting in contrast to other decision modeling notations. An interesting starting point for such an analysis might be a paper by Bock (2015), who has started to identify and compare semantic constructs for decision making in various visual modeling approaches, e.g. decision matrices, decision trees and influence diagrams. In comparison to other approaches, DMN has a strong focus on routine operational decisions and is less suitable to ambiguous, non-routine and novel decision situations (Bock 2015).

Graphic Economy

In comparison to other modeling notations like BPMN, which offer a high number of symbols, DMN can be considered parsimonious because the size of its vocabulary is manageable (4 symbols, 3 types of edges for different requirements and a visual definition for using textual annotations for DRDs (Object Management Group 2016, p. 30)). Based on a theoretical approach analyzing the objects, relationships and properties of the metamodel, the cumulative complexity of DMN was rated “relatively low”, comparable to the modeling standard CMMN, but lower than BPMN, which offers higher expressive power (Hasic et al. 2017). In the authors’ assessment they conclude that “DMN should be simple to learn and understand” (Hasic et al. 2017, p. 69).

Visual Expressiveness and Perceptual Discriminability

The use of visual variables for symbols (position, color, size, texture, shape, orientation, brightness) determines the visual expressiveness of a
notation. Pairwise differences of symbols on visual variables increase perceptual discriminability. In general, DMN uses rectangles to represent decisions and variations of rectangles for other concepts: e.g. for the concept Knowledge Source a “shape with three straight sides and one wavy one”, for Business Knowledge Model a “rectangle with two clipped corners” and for Input Data a “shape with two parallel straight sides and two semi-circular ends” (Object Management Group 2016, p. 31). Requirements differ according to the texture (dotted vs. solid lines) and type of arrowhead. Therefore, Dangarska et al. (2016) have rated visual expressiveness of DMN as low and perceptual discriminability as partly violated. Empirical research is necessary to find out whether low visual expressiveness and discriminability actually lead to problems for users to distinguish symbols.

Semantic Transparency
Semantic transparency is determined by how easily the meaning of a visual appearance can be “inferred from its appearance” (Moody 2009, p. 764). Dangarska et al. (2016) have given the core symbols an opaque score (indicating an “arbitrary relationship between appearance and meaning”), while arrows for Requirements were rated as immediate. Semantic transparency is also relevant for choosing spatial arrangements of elements that ease the comprehension of their relationship. The DMN standard does not give concrete instructions on how to arrange and layout DRDs. When looking at simple DRDs drawn in the standard document (Object Management Group 2016, p. 75), input data and sub-decisions are placed below decisions, business knowledge symbols are placed left and knowledge sources symbols are placed on the left and on the right above decisions. Empirical research could test whether placement of model elements has any effect on the comprehension of DRDs and whether there are positioning guidelines of elements which would result in easier to understand DRDs. In studies on interpreting diagrams with nonsense words “causes were always thought to lie to the left of and above effects” (Winn 1990, p. 155). Thus, placing input data and sub-decisions to the left or above decisions might also be intuitively understandable. However, tree structures expanding from a parent node at the top or from the left are also widely-used conventions, supporting the exemplary spatial arrangement in the standard document (Object Management Group 2016, p. 75). Concerning the layout of decision tables the DMN standard document also gives users freedom of choice and states “a decision table can be presented horizontally (rules as rows), vertically (rules as columns), or crosstab (rules composed from two input dimensions)” (Object Management Group 2016, p. 75). However, the standard is quite clear on the positioning of input columns, which reflect the findings of Winn (1990) on intuitively understandable conventions of spatial arrangements: “In a horizontal table, all input columns SHALL be represented on the left of all output columns. In a vertical table, all the input rows SHALL be represented above all output rows” (Object Management Group 2016, p. 75).

Dual Coding
While text should not be used to distinguish between symbols, it can be wise to supplement graphical information with it (Moody 2009). DMN actively encourages text annotations, using a dotted line and a square bracket. Furthermore, it gives clear advice on how to combine text, e.g. “the label . . . SHALL be clearly inside the shape of the DRD element” (Object Management Group 2016, p. 31). Such a guideline can theoretically grounded on the Gestalt law of common region, which posits “the tendency for elements that lie within the same bounded area to be grouped together” (Palmer et al. 2003, p. 312). However, user evaluations of business process models have shown that readers rate labels placed physically close to symbols equally well to labels placed inside a symbol (Figl 2017).
Cognitive Fit

Moody (2009) suggests the use of different visual dialects for experts and novices as well as for different representational media in order to achieve cognitive fit. This is something DMN does not offer. However, DMN addresses the issue of cognitive fit in the wider sense since a main objective of DMN is to combine decision tables and requirements diagrams (DRDs) to account for the fact that both are well suited to represent different types of information elements for different tasks. Cognitive fit theory (Vessey and Galletta 1991) originates from the observation that graph versus table use is suited for different tasks. Still, it might be advantageous to offer users not only decision tables, but also decision trees as additional visualization to comprehend a complex decision. Vessey and Weber (1986) compared decision tables and decision trees in the context of a programming task and found decision trees to outperform decision tables. Similarly, decision trees were found to be more helpful when used in an investment game than the corresponding decision tables (Subramanian et al. 1992). However, for various comprehension tasks more recent experiments revealed that decision tables performed better than decision trees and textual rules (Huysmans et al. 2011). Overall, more work is needed to clarify inconsistencies of prior research and address whether offering decision tree visualizations in addition to decision tables as specified in DMN might enhance human comprehension.

Complexity Management and Hierarchical Structuring of Decisions

To avoid overloading human working memory with large and complex diagrams, Moody (2009, p. 766) suggests that visual notations should provide mechanisms for modularization and hierarchically structuring. Hierarchical structuring of decisions and modularity are a main purpose of DRG. DMN allows the modeler to split up decisions into different tables and specify their connection. DMN leaves it to the implementations of the modeling tool to show interactive visualizations of Decision Requirements Graphs (DRG) in an efficient way: “For any significant domain of decision-making a DRD representing the complete DRG may be a large and complex diagram. Implementations MAY provide facilities for displaying DRDs which are partial or filtered views of the DRG, e.g., by hiding categories of elements, or hiding or collapsing areas of the network. DMN does not specify how such views should be notated, but whenever information is hidden implementations SHOULD provide a clear visual indication that this is the case” (Object Management Group 2016, p. 35). Decomposition of decisions (splitting decisions into sub-decisions) is relevant for DRDs and as decision tables, as the number of input variables is visually shown in the DRDs as well as part of the decision tables. Mertens et al. (2015, p. 161) note “The decision table representation also has some drawbacks. When the decisions themselves are based on a very large amount of conditions and actions, the readability of a table gets lost. In such cases, the decision table will need to be split in multiple smaller tables to allow them to stay manageable.” There is a long history of literature on decision table design, offering guidance to structure decisions into separate tables, to build decision tables using a stepwise methodology and to avoid table anomalies and unnormalized tables. For an overview of guidelines for decision table, see e.g. CODASYL Decision Table Task Group (1982). A recent tutorial on using DMN by Signavio for instance suggests to split decisions into sub-decisions as soon as a decision has 7 or more inputs. Overall, decomposition in models can lead to two different effects: despite the positive effect of abstraction, which eases comprehension and has lead to the common belief that hierarchically structuring is beneficial for model comprehension, it can also lead to a split-attention effect as readers have to switch between different models or tables (Zugal et al. 2012). Depending on the comprehension task, fully flattened models, respectively decision tables including all input variables may even lead to higher comprehension, as experiments have shown in other modeling domains, e.g. process models (Turetken et al. 2016) or data models (Parsons
While the problem is not new, research for the specific nature of decision modeling is needed. The effects of hierarchical structuring will also differ according to the complexity management and cognitive integration mechanism offered by the modeling tools used. When using table simplification, contraction and normalization, it is also important to consider their effect on repeated use and interactivity of elements; elements that heavily interact cannot be comprehended in isolation and heighten the cognitive load of understanding the decision tables (Sweller 2010). Prior research has demonstrated that different modeling strategies related to minimality and repeated use of elements (e.g. for structuring features in feature trees and using cross-cutting concerns (Reinhartz-Berger et al. 2017)) highly affect model comprehension.

**Cognitive Integration**
Both, homogeneous and heterogeneous integration are relevant for DMN: heterogeneous integration, because it is important for users to understand the bridge DRDs form to different types of visual representations (especially BPMN diagrams and decision logic tables); homogeneous integration, because more than one diagram of the same type (DRD) can depict a DRG. Although the visualisation of DRDs should lower the risk of hidden dependencies, which are relationship between components “such that one of them is dependent on the other, but that the dependency is not fully visible” (Green and Petre 1996, p. 153), understanding interconnected decision tables might get hard and should be investigated in empirical studies.

**Labeling and Naming Conventions**
Labels carry the semantic content. Modeling symbols as decisions, input data or knowledge source as well as input and output columns of the decision tables have to be labeled. For labels of process models, research has already demonstrated that users rate verb-object label styles (e.g. “Determine discount”) for tasks as most useful and least ambiguous (Mendling et al. 2010). Since the top-level decision corresponds to a business rule tasks in a BPMN diagram, a DMN tutorial (Signavio 2017) recommends to use exactly the same label (in verb-object label style). There are other labeling/naming styles as output style (“Discount”) and question style (“Does the customer get a discount?”) and Signavio (2017) give the following advice: “It is best to use the output style for all other decisions, but in some cases the question style is more intuitive than the output style.” If labels get longer as it is the case in the question style, segmentation and visual design of labels gets more critical (Koschmider et al. 2016). However, empirical research testing the actual effects of naming conventions and visual design of labels are still missing for process models, thus their results cannot be directly transferred to DMN and the specifics of DMN demand a separate empirical evaluation anyway. Addressing this challenge requires future empirical research building on experimental designs.

### 3.3 Research Directions on Organizational Benefits of DMN

Decisions that are explicitly defined through DMN and not hardcoded inside organizational decision making processes will likely decrease complexity and hence ease the implementation of business rules and analytic technologies. In this way, DMN might contribute to improved efficiency and effectiveness of organizational decision making, e.g. in terms of increased agility (Jonkers et al. 2013), improved business/IT alignment and increased straight-through processing (Taylor et al. 2013). From an organizational point of view, Lemmens (2015) also emphasize the importance of integration of modeling notions, that organizations utilize and develop, namely of process modeling, information modeling and rules modeling, for their organizational goals and operations for agility.

From another perspective, it is also clear that decision execution efficiency is highly affected by the amount of input data that is required to be collected for business process decisions, which is likewise costly for organizations. However, a recent study by Bazhenova and Weske (2017) proposes a method to reduce cost of the input data
collection process through a prioritization algorithm, which might lead to further organizational benefits.

It is also suggested that employing DMN in organizational decision making might be specifically useful in a setting where business rules change frequently and where decisions have high risk for the operations. Hence, sectors like financial services, insurance, energy, and ICT providers are listed as recommended areas of use. So far, various processes with strong regulatory requirements in the financial industry are currently redesigned and formalized using DMN. Among others, these include the know-your-customer process, which is subject to regulations for anti-money laundering and counter-terrorist financing rules. Other sectors like health care (Combi et al. 2016; Servadei et al. 2017; Wiemuth et al. 2017), disaster management (Horita et al. 2016), retailers or logistics might benefit in a similar fashion.

On the other hand, as DMN allows separation of control flow and decisions, it may also provide cooperation and can be shared between different stakeholders in Collaborative Networks (Biard et al. 2015) and increase solidly the level of beneficiary gains in organizational goals, settings and outcomes, all of which extends and broadens the limits of research within this domain.

DMN besides, is taking a role in information system development field as well. In the recent example of Boumahdi et al. (2016), DMN is utilized for defining decision view of the service design in SOA, which may be extended to create Model Driven Architectures automatically. In a similar way, it is also emphasized that the use of DMN is considerable with other conceptual models used in information system development phases like requirement specification (Kluza et al. 2017) to improve decision logic definition. Thus, benefits of DMN, in parallel to organizational ones, in system development field and how to incorporate in various phases is another area of research to be explored.

Questions on this organizational level relate to how and in which circumstances, companies adopting DMN can realize these proclaimed benefits and what success factors play an important role here. Answering these questions requires further research with diverse empirical research methods and cases in this regard.

4 Conclusion

DMN will change the way how processes are specified and implemented. In this paper, we described its technical foundations of decision table research and its theoretical background of modeling research. We identified research directions for investigating its potential benefits on a technological, individual and organizational level, and in this way clarifying what we know and what we don’t know about DMN. Insights into the way how programmed decisions are specified and implemented together with business process will be a cornerstone of future research into information systems and business process management in the years to come.

References


Bosuuyt J., Gaillly F. (2017) investigating the benefits of modeling business processes with BPMN + DMN. MA thesis, University of Gent, Belgium


CODASYL Decision Table Task Group (1982) A modern appraisal of decision tables. Association for Computing Machinery


This work is licensed under a Creative Commons “Attribution-ShareAlike 4.0 International” license.