



# Adaptive Case Management - A Review of Method Support

Birger Lantow<sup>(✉)</sup>

University of Rostock, Albert-Einstein-Str. 22, 18059 Rostock, Germany  
birger.lantow@uni-rostock.de  
<https://www.wirtschaftsinformatik.uni-rostock.de>

**Abstract.** Knowledge-intensive Processes are difficult to support by traditional workflow oriented Business Process Management approaches. Reasons lie in ad-hoc decisions and unpredictable workflows that come with them. Adaptive Case Management (ACM) is a paradigm for the management of knowledge-intensive processes that has recently drawn attention in industry and the scientific community. This development led to the standardization of CMMN as a notation for process models for the ACM implementation. This study assess the availability of method support for the use of ACM and the fitness of CMMN for fulfilling the modeling requirements in this context based on a systematic literature review. As a result missing method support, CMMN shortcomings as well as suggestions for the implementation of ACM in combination with CMMN are discussed.

**Keywords:** Adaptive Case Management  
Systematic literature review · Knowledge-intensive processes  
Method · CMMN

## 1 Introduction

Nowadays IT support for industrial production processes and well defined business processes has reached a certain maturity. Potentials for differentiation on the market stem from knowledge and the use of knowledge. Work processes in this area - so called knowledge-intensive processes - put some difficulties to the inclusion in Business Process Management (BPM) and thus the process support and control by IT systems. Knowledge-intensive processes are for example characterized by out of order task completion, a high autonomy of the so called knowledge workers, and a hardly predictable information demand. Adaptive Case Management (ACM) is a paradigm for the management of knowledge-intensive processes that has recently drawn attention in industry and the scientific community. Based on supporting IT systems, it allows a data or information centered

process control, ad-hoc changes in process models, and execution at run-time [14]. The industrial interest in prospecting the potential of ACM led to a standardization process under the roof of the Open Management Group resulting in a standard notation for ACM process models called Case Management Model and Notation<sup>1</sup> (CMMN). Several tools for Business Process Management have evolved meanwhile that support this notation standard. Having a paradigm, a notation and certain tool support does not necessarily mean that there is also method support for the use of ACM and CMMN. Bider et al. [3] claimed in 2013 that there is a coherent theory required regarding ACM since it has not been more than a collection of practices so far.

The aim of this study is to assess the extent of method support for ACM that has been defined in the scientific community. Considering the prominence of CMMN a second goal is an assessment of the fitness of the standard to meet ACM requirements. This is done based on a systematic literature review.

In order to provide a frame for this literature review, Sect. 2 gives a short view on method theory. The following Sect. 3 describes the overall process of the literature review including a further specification of the research goals by defining research questions. Additionally, the process and results of paper selection are described. Section 4 deeply analyzes the literature. A summary and outlook is provided in Sect. 5.

## 2 Categories for the Assessment of Method Support

This study uses a meta-model based approach to assess the method support for the creation and handling of process models following the ACM paradigm. Scientific literature regarding method support mainly focuses on notation. This can also be found in the results of the systematic literature review in Sect. 3. However, a method does not consist of a notation and created models only. For example, there need to be procedures in order to extract the knowledge that can be found in the models. Depending on the purpose and the context of method application, a combination of different notations and procedures might be required [7]. In the discipline of method engineering, there are several approaches to describe a meta-model of methods for information system engineering. We take the models by Karagiannis and Khn [12] and by Goldkuhl et al. [7] as a base for our assessment. Both approaches together cover extensively what can be considered as required parts of a method in the information system engineering domain. Though, not all these parts might be defined in a method description.

According to Goldkuhl et al. [7] an important starting point for a method is the perspective which defines problems the method can be applied to and the scope. A method is described as a composition of so called method components that each addresses a sub-problem of the method application. Thus, different notations and procedures can be applied depending on the context. Furthermore, a distinction is made between the notation and the concepts used in a method component. Hence, there is a separation between notation symbols and

<sup>1</sup> <https://www.omg.org/spec/CMMN>.

the semantics. For example, different symbols in different notations can refer to the same concepts. In order control the use of method components, a framework is part of the method model. It helps to determine which method components are to be used when. As already mentioned, modeling procedures are used to acquire knowledge. Since the knowledge originates from the domain experts, cooperation forms are also to be defined as part of a method according to Goldkuhl et al.. This includes the definition of roles and interaction forms.

The method model by Karagiannis and Khn [12] has a stronger focus on Model Driven Development (MDD). While not explicitly considering perspective, framework and cooperation forms, the other method concepts from Goldkuhl et al. are represented more detailed in this model. Modeling procedures are further described by their steps and results. The modeling notation is described by syntax and symbols. As an addition from MDD, Karagiannis and Khn add mechanisms and algorithms to the method model. The latter address aspects of model transformation and analysis.

In order to derive a feasible mapping schema for the assessment of method support for ACM more coarse grained categories have been derived from the introduced method models. Furthermore, a common perspective - Creation and handling of process models for ACM - is assumed and thus not further investigated. The resulting categories are shown in Table 1. Approaches found in the literature review (Sect. 3) are analyzed with regard to them.

**Table 1.** Categories for method support assessment.

Category	Comment
Cooperation forms	Identified roles and collaboration settings for those who are involved in the system engineering process
Concepts and notation	Concepts that will be represented in the models and their representation in a certain notation considering symbols and syntax
Procedures and framework	Steps to be performed in the systems engineering process on different levels of abstraction
Tool support	Tools for model creation, analysis and transformation

### 3 Systematic Literature Review

A systematic literature review has been performed in order to assess the state of research regarding method support for the ACM paradigm. Specifically, the following research questions have been in focus.

- *RQ1*: To what extent are the required parts of a method considered in scientific literature on ACM?
- *RQ2*: What is suggested in scientific literature on ACM with regard to the required parts of a method?
- *RQ3*: How is the fitness of CMMN for ACM evaluated in scientific literature on ACM?
- *RQ4*: What solutions for possible shortcomings of CMMN with regard to ACM are suggested in scientific literature?

As described earlier, CMMN is seen as a broadly used and an accepted standard for process modeling in Adaptive Case Management. CMMN is supported by many commercial and research-based tools in the area. Thus, we set the focus on CMMN and possible CMMN adaptations when discussing ACM notations. The analysis process is oriented along the guidelines for a systematic literature analysis presented by Kitchenham [13] and Webster [19]. The review process is divided into four different parts (see Fig. 1). The first activity is to identify conference series, journals and catalogs that are likely to represent the state of the art of research on the topic of interest. Here, a base set of papers for review is extracted by keyword search. The second step is the exclusion/inclusion of papers based on title and abstract. Then, the remaining papers have to be classified and data has to be extracted with regard to the research questions. The classification is based on the categories for the assessment of method support that have been derived in Sect. 2. The fourth and last step is to analyze the extracted data. The next paragraphs describe the performance of these steps in detail.

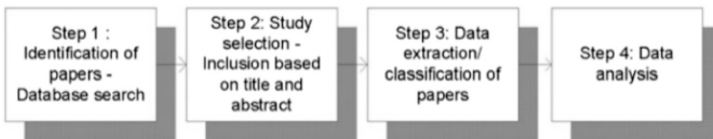


Fig. 1. Systematic literature review process [11].

### 3.1 Identification and Selection of Papers

The first step is the identification of papers dealing with ACM methods and CMMN in a selection of appropriate literature sources. The well-known portals for high quality scientific literature indexing Web of Science/Web of Knowledge and Scopus have been selected as literature sources. In order to include a publisher into the search too, the Elsevier portal has been selected as an additional source. The idea was to check whether a significant contribution to the analysis results can be expected from including publisher portals. Since ACM and CMMN are relatively new topics, the literature search has not been restricted to a specific time period. The following search terms have been applied to publication meta data (Title, Abstract, Keywords):

- "Adaptive Case Management" AND Method
- CMMN AND Method
- "Adaptive Case Management" AND CMMN

The combination of “Adaptive Case Management” and “Method” in the search terms can be directly derived from the research questions. The same is true for “CMMN” and “Method”. The idea of searching for the combination of “Adaptive Case Management” and “CMMN” was that literature combining both should somehow address the use of CMMN in ACM and thus a method for CMMN model handling.

The initial search resulted in a combined set of 38 papers matching at least one of the search terms (see Table 2). First tries with different search terms resulted in a large amount of irrelevant papers. For example the combination “ACM” and “Method” had more than 48,000 hits because a lot of abstracts contained references to the Association of Computing Machinery (ACM). Therefore, only the described search terms have been used. Regarding the different portals that have been used for the search, there has been a high redundancy. Most of the results by Web of Science and Elsevier have also been found at Scopus. Only three added to the total number of candidate papers.

Paper selection singled out papers that did not deal with ACM at all. Furthermore, only those have been selected that considered at least one of the categories for method support assessment. If a paper was discussing an ACM related notation, a reference to CMMN was mandatory for inclusion. The selection process ended with 13 papers that have been used for further analysis.

**Table 2.** Paper identification and selection.

Source	# Identified	# Selected	Selected papers
Scopus	35	13	[1–6, 8–10, 14–16, 18]
Web of science	12	5	[1, 2, 4, 6, 16]
Elsevier	9	2	[14, 18]
Total (no duplicates)	38	13	[1–6, 8–10, 14–16, 18]

## 4 Data Extraction and Analysis

In the following, the findings with regard to the research questions formulated for the literature analysis are discussed. This analysis is based on the 13 papers found.

**Table 3.** Data collection results.

Source	Cooperation forms	Concepts and notation	Procedures and framework	Tool support
Benner-Wickner et al.: Supporting adaptive case management through semantic web technologies [1]		X		X
Bider: Towards Process Improvement for Case Management [2]		X		
Bider et al.: Adaptive Case Management as a Process of Construction of and Movement in a State Space [3]		X		X
Blaukopf and Mendling: An Organizational Routines Perspective on Process Requirements [4]		X		
Buksh et al.: Understanding modeling requirements of unstructured business processes [5]		X		
Cano et al.: An adaptive case management system to support integrated care services: Lessons learned from the NEXES project [6]		X	X	X
Hauder et al.: Empowering End-Users to Collaboratively Structure Processes for Knowledge Work [8]	X	X	X	X
Hinkelmann: Business process flexibility and decision-aware modeling- The knowledge work designer [9]		X		X
Hinkelmann et al.: The knowledge work designer-Modelling process logic and business logic [10]		X		X
Kurz et al.: Leveraging CMMN for ACM [14]		X		
Marin et al.: Data Centric BPM and the Emerging Case Management Standard: A Short Survey [15]		X		X
Routis et al.: Using CMMN to Model Social Processes [16]		X		
Wang and Traore: DEVS-based case management (WIP) [18]		X		X

#### 4.1 RQ1: Extent of Method Support Consideration

The mapping of the found papers to the categories for method support assessment reveals a strong focus on “Notation and Concepts” in scientific literature

when it comes to ACM methods. This can be seen in Table 3. All sources refer to notation or concepts while the categories “Cooperation Forms” and “Procedure and Framework” attract little interest. Many of the papers also refer to tool support for ACM.

## 4.2 RQ2: Suggestions for Method Support

With regard to “Cooperations Forms” there is a common understanding of a role “knowledge worker” who is the addressee of ACM solutions. However, how the knowledge worker is involved in the creation of process models that are the base for process execution in ACM systems is not addressed. Although Cano et al. [6] describe the process of model creation and adaptation, they remain vague regarding the question who is doing what here. The most evident occurrence of cooperation between different roles can be found in Hauder et al.. Here an end-user (knowledge worker) enters data that can be used by modeling experts. However, while end-users work collaboratively in a wiki, cooperation of and with modeling experts is not further described.

The category “Notation and Concepts” will be evaluated thoroughly when *RQ3* and *RQ4* are discussed. In principle, four general suggestions can be derived from literature:

1. Using a reduced set of concepts in order to allow the end-user/knowledge worker to contribute to the modeling process [3,8]
2. Extending the CMMN notation with additional concepts [5]
3. Amending CMMN model by concepts from other standards like BPMN<sup>2</sup> and DMN<sup>3</sup> [9,10] or SBVR<sup>4</sup> [1]
4. Transforming CMMN models into other notations for further processing [2,18]

The other authors just emphasize on possible CMMN shortcomings without discussing solutions or naming CMMN compatible concepts that are used in their methods or tools.

There is also little information regarding “Procedures and Framework” in the literature. Hauder et al. roughly describe a framework and procedures as a two step process. In the first step, end-users enter process relevant information collaboratively in a wiki. They are using a textual representation in order to specify a process type, tasks to be performed and relevant attributes for task fulfillment. In a next step modeling experts create CMMN based process models (case plan models) as templates for process execution. Adaptation and Feedback is not further considered. Cano et al. [6] describe a general ACM process that also contains phases of process modeling and of model adaptation at run-time. However, concrete procedures to elicit model contents are not described. The process is depicted in Fig. 2. It shows the process of typical clinical case handling in combination with ACM. Starting with the “Case Identification”, it is

<sup>2</sup> <https://www.omg.org/spec/BPMN/>.

<sup>3</sup> <https://www.omg.org/spec/DMN/>.

<sup>4</sup> <https://www.omg.org/spec/SBVR/>.

determined whether a certain patient is eligible for a case management based treatment. In the next step the “Case Evaluation”, parameters are assessed, providing information for the selection of possible work plans for further treatment. The work plan is determined in the “Work Plan Definition” step (shaded gray). This is the point where process models are created in form of work plan templates and instance (patient) based work plans. Precondition for this modeling step is a library of possible tasks that can be performed in the work plan and a library of work plan templates. The step of “Work Plan Definition” can be applied several times during work plan execution in order to allow situation based adaptations. This is triggered by events and follow-ups. The management of the case instance ends with the “Discharge” step - the patient leaves.

For the category “Tool Support”, the literature can be divided into two groups: (1) sources describing workflow management systems that support ACM and (2) sources describing solutions for process model transformation and/or analysis. Several authors from the first group present systems that have been developed in their research groups: Bider et al. with the iPB-System [3], Cano et al. with the ICS-system developed in the NEXES project [6], and Hauder et al. with the Darwin Wiki [8]. Though not being exhaustive, Marin et al. give an overview of commercial ACM capable systems [15]: FLOWER from Pallas Athena, IBM Case Manager, Cordys Case Management. To the second group of authors belong Wang and Traore who discuss the transformation of CMMN models to DEVS models in order to allow simulation and formal verification. with the Knowledge Work Designer, Hinkelmann et al. provide a tool that allows the integration of CMMN, DMN, and BPMN [9,10]. Other authors that would have been assigned to this group did not refer to CMMN and aren’t considered further.

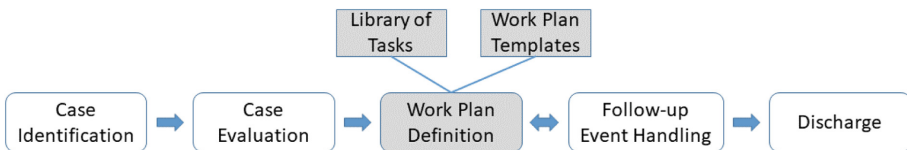


Fig. 2. Case management process according to Cano et al. [6].

### 4.3 RQ3: Fitness of CMMN for ACM

Two general approaches exist in literature when evaluating CMMN for ACM. The first approach considers the method perspective on CMMN. Hence, roles in the modeling and the modeling process itself are taken into account. Thus, Bider et al. [3], Hauder et al. [8], and Kurz et al. [14] see the problem of understandability of process models for end-users/knowledge workers that are involved. Furthermore, Kurz et al. address the use of CMMN in the ACM process similar to the process shown in Fig. 2, where the “Work Plan definition” step



includes model changes at runtime and the usage as well as the improvement of model templates across cases. This reflects the ACM principle of adaption [14]. Since model adaption is primarily a matter of the modeling process, it is not further discussed with regard to notation.

The second approach for the evaluation of CMNN for ACM is to assess which concepts need to be considered and thus modeled when implementing ACM. The resulting concepts can be compared to those defined in CMMN. Hinkelmann [9], Bukhsh et al. [5], and Kurz et al. [14] take this approach. There is a general understanding, that ACM addresses mainly unstructured, knowledge-intensive processes. This implies, that there are also structured parts of the processes that need to be modeled. Besides the so called process logic also the business logic hence data and the decision processes based on that data need to be considered [9]. This is also reflected by the ACM principles of data-centricity and goal-orientation as formulated by Kurz et al. [14]. Progress towards process goals should be somehow reflected in the process data. Hinkelmann [9] argues that business logic (data) in knowledge-intensive processes may be structured or unstructured as well, similar to the situation with the process logic. Further modeling requirements can be derived from the ACM principles of collaboration and integration of resources [14]. This requires a role concept and concepts for traceability. Summarizing the discussion, three fields of notation requirements for ACM can be derived: (1) Model support for process logic, (2) Model support for business logic, and (3) Model support for Collaboration. Table 4 collects the particular requirements from the literature sources and assigns them to the three fields. Furthermore, their fulfillment by CMMN is evaluated. The judgment is based on the analyzed sources and the current CMMN 1.1 standard definition. In the following detailed discussion, existing CMMN concepts are set in `typewriter` font.

Table 4 shows a good support for modeling the process logic of unstructured processes. CMMN does not allow a direct modeling of the temporal order of tasks. This can be done implicitly using the `Sentry` concept of CMMN which allows the definition of preconditions for task execution. Therefore, a partial support is indicated in Table 4 for requirement P1. There are three concepts defined in CMMN that allow different levels of task granularity (Requirement P2). The `Stage` concept may contain `Tasks` and `Stages` that are subject to `Stage`-specific process execution rules. Furthermore, the concept of a `CaseTask` refers to other CMMN process models while the concept of a `ProcessTask` refers to structured process models in BPMN, XPDL<sup>5</sup>, or BPEL<sup>6</sup>. Thus, a more detailed specification of these tasks can be modeled. Requirement P2 is considered to be fulfilled. Out of order task execution (Requirement P3) is inherently supported by CMNN because the order of task execution is generally not prescribed by CMMN. Defining required and optional tasks as well as the possible re-execution of tasks (Requirements P4 and P5) is possible using the `PlanItemControl` concept of CMMN. Bukhsh et al. [5] suggest undo tasks as a special task type (Requirement P6). This concept is not available in CMNN. Furthermore, they suggest

---

<sup>5</sup> <http://www.xpdl.org/>.

<sup>6</sup> <http://docs.oasis-open.org/wsbpel/2.0/>.

to model an execution relation between tasks which implies a required parallel execution of tasks. This is justified by the collaborative nature of knowledge-intensive processes (Requirement P7). The parallel execution concept is not available in CMMN as well.

Discussing Business logic representation in CMMN covers several aspects. First, the principle of data-centricity requires the possibility to create a data or information model. With the **CaseFile** concept, this can be integrated in the process model. The **CaseFile** contains all information objects related to the process - it is a collection of **CaseFileItems** for a process model. A **CaseFileItem** can be any piece of information - structured or unstructured (Requirement B1). CMMN allows to define cardinalities of and simple relations between **CaseFileItems**. Each **CaseFileItem** comes with a predefined set of possible state-changes throughout the information life-cycle (**CaseFileItemTransitions**, Requirement B2). Similarly, there is a predefined set of transitions for processes and their elements (**PlanItemTransitions**). CMMN provides the **Property** concept which may be used to assign attributes and their types to **CaseFileItems**. A concept for data version control (Requirement B3) is absent. An alignment of information in the **CaseFile** to the executed tasks is provided by the **CaseParameters** concept which can be used to define **CaseFileItems** as in- and outputs of a task (Requirement B4). Generally, CMMN does not provide a graphical notation for information modeling. This can be seen as a drawback regarding understandability of the notation.

The CMMN information model forms the base for the definition of business rules and decisions. Taking the common definition of business rules to provide general guidelines for organizational behavior, decision models also form business rules (e.g. [1]). However, due to their importance in ACM according to the literature and their inherent complexity they will be handled separately. Different types of business rules can be distinguished. Sandkuhl et al. define the following types in [17]: (1) Derivation Rules, (2) Event-Action rules, and (3) Constraint rules. The first type - Derivation Rules - describes the derivation of new information from the existing information base. Hence, they describe information that is implicitly available by applying the business rules to existing information objects. Generally, CMMN does not provide concepts for information object manipulation. The **ParameterMapping** is an exception. It provides means to derive parameter values for sub-process execution from **CaseFileItems**. However, this is not an appropriate way to define business rules. Furthermore, CMMN allows the definition of **Decisions** within a **DecisionTask** using DMN. Thus, new information objects or information object values can be derived as decision results based on decision modeling. The second type of business rules - Event-action Rules - concerns the invocation of tasks. This purpose is fulfilled by the **Sentry** concept in CMMN. Pre- and post-conditions of task execution can be defined based on the predefined transitions of process elements in the information model, timer-events, and user-defined events. Whereas user-defined event specification is limited to event names, the **Sentry** may also contain complex expressions that include elements of the types **CaseFileItem** and **Property**.

Nevertheless, no expression language is specified for CMMN. The conditions defined by a **Sentry** are required to be fulfilled in order to allow task execution, but are only sufficient for task execution in the case of automated tasks. The third type of business rules – Constraint Rules – defines constraints for the integrity of the enterprise information and for the execution of tasks. While the execution of tasks can be constrained using the **Sentry** concept, no concept is available for information related constraints except for cardinalities. In summary, a partial support for business rule definition (Requirement B5) is seen for CMMN.

As already discussed, the concepts of **Decision** and **DecisionTask** allow the creation of decision models (Requirement B6). The results of decisions at runtime (Requirement B7) are not considered in the information model of CMMN. Case progress (Requirement B8) can partially be described using the CMMN concept of a **Milestone**. A **Milestone** stands for a desired state in the process execution and hence for progress and achieved objectives (Requirement B9). While the **Milestone** concept is available, it remains open how milestones can be connected to process information that allows the derivation of the current process state. Thus, these requirements are only partially supported.

Having a look into collaboration support, first the existence of a **Role** concept (Requirement C1) is mandatory. It is defined in CMMN. **Tasks** can be assigned to **Roles**. Nevertheless, this is not part of the visual notation of CMMN and an organizational model does not exist. An assignment of roles to individuals is not part of the notation as well. Thus, making individuals responsible for certain tasks (Requirement C2) is not supported. Due to CMMN's simple information model, there is no concept to specify authorizations for data access (Requirement C3).

#### 4.4 RQ4: Notation Suggestions

The general suggestions for addressing the shortcomings of CMMN have already been introduced in the discussion of *RQ2*. The first group of suggestions – reducing the number of used concepts – stems from those authors that criticized the understandability of CMMN for end-users. Hauder et al. [8] reduce the concepts that end-users are required to use down to three: (1) **Task**, which is also present in CMMN, (2) **Attributes**, which correspond to **CaseFileItem** or **Property**, and (3) **Type**, which identifies the case and allows the generation of case templates. Further specification of process models is done by modeling experts based on the information obtained from end-users. Bider et al. [3] reduce the process model to a space of desired states on a given data structure. The data structure corresponds to the **caseFile** concept in CMMN. The closest CMMN concept to a **State** is a **Milestone**. In combining data structure and states, this approach addresses data-centricity better than CMMN.

Following the discussion of the CMMN limitations with regard to required CMMN concepts leads to suggestions 2 and 3 (see *RQ2*) which means the use of additional concepts either by own additions or by reference to other existing standards. Bukhsh et al. [5] decided to define own concepts and their notation.

**Table 4.** Requirements fulfillment of CMMN

Process logic		
ID	Requirement description	Fulfillment
P1	Temporal order of tasks [5,9,14]	Partially
P2	Different levels of task granularity [5,14]	Yes
P3	Tasks out of order [5,9,14]	Yes
P4	Optional and required tasks[5]	Yes
P5	Re-executable tasks[5]	Yes
P6	Undo tasks[5]	No
P7	Collaboration between tasks[5]	No
Business logic		
ID	Requirement description	Fulfillment
B1	Integrate unstructured documents and structured data [5,9,14]	Yes
B2	Model information life-cycle [5]	Yes
B3	Data version control [5]	No
B4	Align data with process [5]	Yes
B5	Define business rules [1,5,9]	Partially
B6	Model decisions [5]	Yes
B7	Capture decisions [5]	No
B8	Show Case Progress [14]	Partially
B9	Expression of case objectives [14]	Partially
Collaboration		
ID	Requirement description	Fulfillment
C1	Support a role concept [5,9,14]	Partially
C2	Show individual responsibilities [9,14]	No
C3	Data authorization[5]	No

In comparison with the existing elements of the current CMMN version, the following additions would be made:

- Concept and symbol of a Collaborative sub-process (Requirement P7)
- Concept and symbol of an Undo task (Requirement P6)
- Symbol of a Role (Requirement C1)
- Symbol and notation of a Business Rule (Requirement B5)

Hinkelmann et al. [9,10] use a combination of CMMN and BPMN for the description of process logic in their modeling approach. This results in BPCM (Business Process and Case Management Notation) which allows an extended visualization of structured process parts using BPMN (Requirement P1) as well as of role assignments to tasks using the Lane concept from BPMN (Requirements C1

and partly C2). In the business logic domain, a graphical notation for information objects is provided.

Benner-Wickner et al. suggest to use SBVR for extended business rule support [1] (Requirement B5).

## 5 Conclusion and Outlook

the aim of this work was the assessment of the method support for ACM and the fitness of CMMN in this context. This has been further detailed to the four research questions that will shortly be answered in the following:

- *RQ1*: To what extent are the required parts of a method considered in scientific literature on ACM?  
Generally, a strong focus lies on notations and used concepts. This is considered in connection with process execution systems that use process models in the respective notations. In contrast, cooperation forms and modeling procedures receive little attention.
- *RQ2*: What is suggested in scientific literature on ACM with regard to the required parts of a method?  
Except for suggested notations in connection with tool support, there are two roles distinguished that need to be addressed differently in the modeling process - end-users and modeling experts. Furthermore, a rough modeling process with two steps specific to the two roles is described [8]. The paper of Cano et al. [6] suggests a general process for ACM implementation in a clinical context. This also includes the definition of tasks for run-time adaptation of the process model instance.
- *RQ3*: How is the fitness of CMMN for ACM evaluated in scientific literature?  
One key point mentioned in the analyzed papers is the lack of understandability of CMMN models for end-users. Another point are the limitations regarding the modeling of business logic and collaboration.
- *RQ4*: What solutions for possible shortcomings of CMMN with regard to ACM are suggested in scientific literature on ACM?  
As a solution for the first shortcoming of CMMN the reduction of used model concepts has been suggested. As a solution for the limited representation capabilities for important ACM concepts the extension by new concepts or concepts from other notations is discussed.

Overall, the current situation is characterized by a lack of method support for the modeling process. Most studies emphasize on the models and their usage in process execution but not on the model creation and adaption. Furthermore, it seems that collaboration aspects and data-centricity need to be addressed by future notation standards in the context of ACM. However, considering the mentioned problem of understandability, an extension of the available concepts for modeling will increase the demand for a method support regarding modeling procedures and frameworks. Different roles with different knowledge are likely to be involved in modeling.

The results of this study might be biased by the limited amount of analyzed papers. However, using the method of a systematic literature review, it is assumed that a representative cross section of scientific literature has been considered. Thus, there is a good possibility of generalization.

## References

1. Benner-Wickner, M., Koop, W., Book, M., Gruhn, V.: Supporting adaptive case management through semantic web technologies. In: Reichert, M., Reijers, H. (eds.) BPM 2015. LNBIP, vol. 256, pp. 65–77. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-42887-1\\_6](https://doi.org/10.1007/978-3-319-42887-1_6)
2. Bider, I.: Towards process improvement for case management. In: Reichert, M., Reijers, H. (eds.) BPM 2015. LNBIP, vol. 256, pp. 96–107. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-42887-1\\_9](https://doi.org/10.1007/978-3-319-42887-1_9)
3. Bider, I., Jalali, A., Ohlsson, J.: Adaptive case management as a process of construction of and movement in a state space. In: Demey, Y.T., Panetto, H. (eds.) OTM 2013. LNCS, vol. 8186, pp. 155–165. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-41033-8\\_22](https://doi.org/10.1007/978-3-642-41033-8_22)
4. Blaukopf, S., Mendling, J.: An organizational routines perspective on process requirements. In: Teniente, E., Weidlich, M. (eds.) BPM 2017. LNBIP, vol. 308, pp. 617–622. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-74030-0\\_48](https://doi.org/10.1007/978-3-319-74030-0_48)
5. Bukhsh, Z.A., van Sinderen, M., Klaas, N.S., Quartel, D.: Understanding modeling requirements of unstructured business processes. In: ICETE 2017 - Proceedings of the 14th International Joint Conference on e-Business and Telecommunications, vol. 2 (2017)
6. Cano, I., et al.: An adaptive case management system to support integrated care services: lessons learned from the NEXES project. *J. Biomed. Inform.* **55**, 11–22 (2015). <https://doi.org/10.1016/j.jbi.2015.02.011>
7. Goldkuhl, G., Lind, M., Seigerroth, U.: Method integration: the need for a learning perspective. *IEE Proc. Softw.* **145**(4), 113 (1998). <https://doi.org/10.1049/ip-sen:19982197>
8. Hauder, M., Kazman, R., Matthes, F.: Empowering end-users to collaboratively structure processes for knowledge work. In: Abramowicz, W. (ed.) BIS 2015. LNBIP, vol. 208, pp. 207–219. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-19027-3\\_17](https://doi.org/10.1007/978-3-319-19027-3_17)
9. Hinkelmann, K.: Business Process flexibility and decision-aware modeling—the knowledge work designer. In: Karagiannis, D., Mayr, H., Mylopoulos, J. (eds.) Domain-Specific Conceptual Modeling, pp. 397–414. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-39417-6\\_18](https://doi.org/10.1007/978-3-319-39417-6_18)
10. Hinkelmann, K., Pierfranceschi, A., Laurenzi, E.: The knowledge work designer-modelling process logic and business logic. In: Lecture Notes in Informatics (LNI). In: Proceedings - Series of the Gesellschaft für Informatik (GI), p. 255 (2016)
11. Ivarsson, M., Gorschek, T.: Technology transfer decision support in requirements engineering research: a systematic review of REj. *Requir. Eng.* **14**(3), 155–175 (2009)
12. Karagiannis, D., Kühn, H.: Metamodeling platforms. In: Bauknecht, K., Tjoa, A.M., Quirchmayr, G. (eds.) EC-Web 2002. LNCS, vol. 2455, pp. 182–182. Springer, Heidelberg (2002). [https://doi.org/10.1007/3-540-45705-4\\_19](https://doi.org/10.1007/3-540-45705-4_19)

13. Kitchenham, B.: Procedures for performing systematic reviews. Keele, UK, Keele University **33**(2004), 1–26 (2004)
14. Kurz, M., Schmidt, W., Fleischmann, A., Lederer, M.: Leveraging CMMN for ACM. In: Unknown (ed.) Proceedings of the 7th International Conference on Subject-Oriented Business Process Management - S-BPM ONE 2015, pp. 1–9. ACM Press, New York, New York, USA (2015). <https://doi.org/10.1145/2723839.2723843>
15. Marin, M., Hull, R., Vaculín, R.: Data centric BPM and the emerging case management standard: a short survey. In: La Rosa, M., Soffer, P. (eds.) BPM 2012. LNBIP, vol. 132, pp. 24–30. Springer, Heidelberg (2013). [https://doi.org/10.1007/978-3-642-36285-9\\_4](https://doi.org/10.1007/978-3-642-36285-9_4)
16. Routis, I., Nikolaidou, M., Anagnostopoulos, D.: Using CMMN to model social processes. In: Teniente, E., Weidlich, M. (eds.) BPM 2017. LNBIP, vol. 308, pp. 335–347. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-74030-0\\_25](https://doi.org/10.1007/978-3-319-74030-0_25)
17. Sandkuhl, K., Stirna, J., Persson, A., Wißotzki, M.: Enterprise Modeling: Tackling Business Challenges with the 4EM Method. TEES. Springer, Heidelberg (2014). <https://doi.org/10.1007/978-3-662-43725-4>
18. Wang, S., Traoré, M.K.: DEVS-based case management (WIP). Simul. Ser. **46**(4) (2014)
19. Webster, J., Watson, R.T.: Analyzing the past to prepare for the future: writing a literature review. MIS q., xiii–xxiii (2002)