



# Empirical evaluation of CMMN models: a collaborative process case study

Ioannis Routis<sup>1</sup> · Mara Nikolaidou<sup>1</sup> · Dimosthenis Anagnostopoulos<sup>1</sup>

Received: 16 November 2018 / Revised: 29 March 2020 / Accepted: 27 April 2020 / Published online: 9 June 2020  
© Springer-Verlag GmbH Germany, part of Springer Nature 2020

## Abstract

Case Management Model and Notation (CMMN) was introduced by the Object Management Group as an alternative modeling language, targeting human-centric processes characterized by lack of structure and agility. However, although it is adequately supported by well-known process management tools, CMMN applicability as a modeling language is being questioned in practice. In this work, an empirical evaluation of CMMN models is presented, through a real-world case study where CMMN has been used for the analysis and implementation of a collaborative process by independent groups of process engineers. Their experience is being discussed, based on their modeling perspective. The produced models in the analysis and implementation phase are evaluated, using pre-existing metrics customized for CMMN. Based on the experience of engineers, CMMN applicability is evaluated, highlighting aspects in which its application might be limited, that should be addressed.

**Keywords** Process modeling · Collaborative processes · Modeling perspectives · Case Management Model and Notation · Evaluation metrics · Empirical evaluation

## 1 Introduction

Collaborative processes are recursive ones, relatively complex, normally developed and shared among a group of stakeholders with different goals and skills, working together, sharing knowledge, learning and building consensus [6]. Business Process Model and Notation (BPMN), well known for its structural philosophy, is not efficient for modeling such processes, as BPMN work-flows are designed to be strict, non-adaptive to changes as well as not supportive to decision-making and collaboration [5,25,30,38]. On the other hand, there are alternative-to-BPMN languages proposed to model collaborative processes [2,5,8,9,40]. One of the main differences between languages like BPMN and such efforts is the paradigm shift from prescriptive to declarative [7].

However, none of these languages has been established as a standard. To this end, the Case Management Model and Notation (CMMN) was introduced by the Object Management Group (OMG), to facilitate and support organizations that would prefer to view their processes as cases, where participants exchange data and ideas to fulfill a specific goal, while there are no strict rules governing their interaction [20]. According to OMG, CMMN may aid in the decision making process through suggestions, yet keeps humans firmly in the driver's seat. CMMN is centered around living information and relationships, while BPMN is centered around a-priori defined activity sequences. Using an event-centered approach and the concept of a case file, CMMN expands the boundaries of what can be modeled with BPMN, including less structured work efforts and those driven by knowledge workers. It is anticipated that using a combination of BPMN and CMMN allows users to cover a much broader spectrum of work methods [20].

When requested to consult a public organization on exploring a collaborative process and develop the appropriate software to support it, we decided to employ CMMN following OMG guidelines. ReWeee Initiative employed by Appliances Recycling SA, alongside with more than twenty collaborators both Greek and foreign, aims at promoting electronic product exchange and consequently reduce electronic

---

Communicated by Rainer Schmidt and Jens Gulden.

✉ Ioannis Routis  
i.routis@hua.gr  
Mara Nikolaidou  
mara@hua.gr  
Dimosthenis Anagnostopoulos  
dimosthe@hua.gr

<sup>1</sup> Harokopio University of Athens, Athens, Greece

waste. Participating in this initiative, we were assigned the design and implementation of a collaborative platform to promote electronic and electrical equipment (EEE) exchange. For this purpose, we employed CMMN to model the EEE exchange collaborative process. Despite the fact that there is some uncertainty about CMMN applicability in real-world cases [23,41], we explored its potential for implementation purposes as well.

Our first task constituted of the user requirement analysis phase, where more than twenty project stakeholders argued on the way exchange should be performed. The attempt to provide them with a CMMN model of the process to promote discussion was presented in [32]. Two different groups worked independently to construct a CMMN model for EEE exchange process for analysis purposes, and their experience was recorded. Their effort resulted in two different models, each following a different perspective, one analytic and one abstractive, although both models conformed to CMMN semantics. The expectations of the two different groups regarding CMMN as a modeling language were recorded, along with challenges faced when using it.

In this paper, we extend the work presented in [32], commenting on the experience obtained during the overall process of analysing and implementing the EEE exchange process using CMMN. By presenting this experience, we aim to explore CMMN applicability in modeling collaborative processes in both the analysis and the implementation phase of a project. Considering the aforementioned difference of the two alternative modeling perspectives and corresponding CMMN models between the two groups, we retained them in the implementation phase as well. Both groups were given an open-source CMMN platform to implement the model they have constructed in the analysis phase. They were given the freedom to alter the model as they found fit to make it more efficient for implementation purposes. Their experience was recorded, codified in specific queries for both the user analysis and implementation phase of the project. Emphasis was given on the modeling language itself and its execution features, as prescribed in CMMN standard, and not the specific features of existing tools or rather the lack of them. As the adoption of CMMN standard is in its initial stage, all CMMN execution engines are expected to fully comply to CMMN standard and provide prescribed functionality in short period of time. Propositions for future improvement focus on the language itself and not specific tools supporting it.

As the two analysis models, although valid in terms of CMMN constructs, differ significantly between them, a quantitative comparison between them in both the analysis and implementation phase took place to explore whether they would align during implementation. To do so, existing

CMMN model evaluation metrics regarding their complexity [19] and expressiveness [7], were employed. Additionally, the comparison of the results led to observations regarding how modeling philosophy was altered during the shift from analysis to implementation phase. Combined with the recorded experience of process engineers, we resulted in reflections on CMMN applicability for modeling collaborative processes.

The rest of the paper is structured as follows. In Sect. 2, a brief introduction to CMMN language is provided alongside with a discussion on existing work on CMMN evaluation and collaborative processes modeling. In Sect. 3, ReWeee case study is presented and the setup of the empirical research procedure followed to explore and record the experience of process engineers is explained. In Sect. 4, the modeling experience through the two perspectives identified in [32] is briefly discussed. Sect. 5 focuses on implementation experience. The effort of the two modeling groups to implement their models is being also discussed through queries about their experience. In Sect. 6, the evaluation of CMMN applicability is discussed, while the models created are being evaluated as well. The evaluation questions are answered based on the experience recorded in the previous sections. Conclusions and future work reside in Sect. 7.

## 2 Background: related work

### 2.1 Modeling of collaborative processes

Collaborative processes are recursive ones, where two or more people or businesses work together aiming on common goals [25]. Within the collaborative process, management information is driven by social, technological, scientific, and inter-disciplinary dependencies. Collaborative processes are therefore difficult to handle. [6].

Business Process Model and Notation (BPMN), well known for its structural philosophy, is not efficient for modeling such processes, as BPMN work-flows are designed to be strict, non-adaptive to changes as well as not supportive to decision-making and collaboration [5,25,30,38]. On the other hand, there are alternative-to-BPMN languages proposed to model collaborative processes [2,5,8,9,40]. One of the main differences between languages like BPMN and such efforts is the paradigm shift from prescriptive to declarative [7].

More specifically, in [5] a role-based framework for modeling collaborative processes is proposed, while in [40], a modeling approach for collaborative business processes is being introduced. Additionally, in [9], a support system for

ad hoc and collaborative processes is being presented. In [8] an event-based approach based on Complex Event Processing (CEP) was proposed to deal with collaborative processes. A similar attempt is described in [1]. The need to translate models of collaborative processes, created using such tailored languages to executable BPMN model for implementation purposes has been identified in [25], due to the lack of any standardisation.

CMMN was introduced by OMG in 2015 as an effort to provide a standardized, executable modeling language, as an alternative to BPMN [20]. Its suitability for modeling collaborative processes has been proclaimed, though it should be explored in practice [31].

## 2.2 CMMN standard

The focus of the CMMN specification is the notation, the meta-model, the interoperability between tools, and minimum execution semantics [18]. The main objective of CMMN is to define a common meta-model and notation for modeling and graphically expressing a Case. A Case involves actions taken regarding a subject in a particular situation to achieve a desired outcome. Traditional examples are cases that refer to legal and medical working environments, where a legal case involves the application of the law to a subject in a certain fact situation, and a medical case involves the care of a patient in the context of a medical history and current medical problems. The subject of a case may be a person, a legal action, a business transaction, or some other focal point around which actions are taken to achieve an objective. The situation commonly includes data that inform and drive the actions taken in a case [22].

There are two phases for each executable case. First, during design-time, business analysts prepare the case execution by modeling the case. Once a case has started to being executed, the case is in run-time. In this phase, the case workers are working on achieving the case objectives [17]. A CMMN model primarily comprises of the case items projected in Table 1.

Case management is concerned with determination of which tasks are applicable, or which follow-up (discretionary) tasks are required, given the state of the case. Decisions and flow may be controlled by events or new facts that continuously emerge during the course of the case, such as the receipt of new documents, completion of certain tasks, or achieving certain Milestones. Individual tasks that are planned and executed in the context of the case might be predefined procedural processes in themselves, but the overall case cannot be orchestrated by a predefined sequence of tasks. Finally, the meta-model and notation are used to express a case model in a common notation for a particular

type of cases, and the resulting model can subsequently be instantiated for the handling of a particular instance of a case [22].

## 2.3 Using and evaluating CMMN language

CMMN is currently considered an alternative-to-BPMN language [22]. Several applications of Case Management, can be identified, including among others, patient treatment and medical diagnosis in healthcare, mortgage processing in banking and application and claim processing in insurance [22]. Very few of these applications can be found in the literature, that use CMMN in real-world scenarios. Such an attempt is the one of Wiemuth et al. [41], where CMMN, combined with BPMN and DMN [21] has been applied in modeling real-world operations taking place in two hospitals in Germany. Another work, the one of Herzberg et al. [13], applies CMMN “to tackle the need of flexibility and transferability of clinical pathways between hospitals” [13]. Lantow et al [4] discuss its application in the social sector.

Having applied CMMN on real-world scenarios, the next step one should take is to evaluate such an attempt. There are plenty of research works aiming to the evaluation of a methodology or a modeling language, like [29] and [42], that attempt to evaluate BPMN. In order to evaluate the application of a methodology upon a field of study, what it is required, is the identification of evaluation metrics for this methodology. Regarding BPMN, there are many attempts for defining metrics, including [11,12,15,17]. As far as CMMN is concerned, recent works attempt to identify metrics for models evaluation, including [7,19]. With these metrics as a basis one could attempt to evaluate the application of solely CMMN in a real-world case study.









In [14,43], the usage of CMMN is compared with the one of BPMN based on a specific case study.

Recently, CMMN was used as a language for modeling knowledge and collaborative work, i.e., knowledge-intensive and collaborative processes [10,37]. Previous work by the authors examines CMMN for modeling social and collaborative processes [31,32].

## 3 Modeling a collaborative process with CMMN

In order to explore the applicability of CMMN for the analysis and implementation of collaborative processes, we present our experience of using CMMN in a real-world case study. As identified in [32], there is a differentiation in the perspectives between different groups of process engineers. This is not an unusual issue at the requirements phase of a

**Table 1** CMMN elements, notation and description

Name	CMMN notation	Description
Case Plan Item		A case plan item contains all the case elements that are involved representing the content of the case as well as the way to process and resolve the case
Task Plan Item		Tasks describe activities that can be executed during the run-time phase. Four types of tasks are supported: human (performed by a knowledge worker), process (to embed a process), decision (to embed a decision) and case (to embed other cases)
Case File Item		A Case File Item represents a piece of information of any nature, ranging from unstructured to structured, and from simple to complex. In knowledge-intensive work, documents are typical outputs of tasks or stages
Stage Plan Item		Stages are logical containers of tasks to be performed within the course of a case. They allow structuring a case hierarchically
Milestone Plan Item		A Milestone represents an achievable target, defined to enable evaluation of progress of the case. No work is directly associated with a Milestone, but completion of set of tasks typically leads to achieving a Milestone of the case
Event Listeners		An Event Listener captures events, which are things that happen during a case. Events may trigger, for example, the enabling, activation, and termination of stages and tasks, or the achievement of milestones
Sentries		Sentries allow defining logical dependencies between tasks and/or events, watching out for important situations to occur. Sentries also represent a combination of conditions and events that define the sequence of tasks to be implemented [31]
Discretionary Items		These identify an item, of which instances can be planned, to the discretion of a case manager[7,17]

project [3], especially when it comes to collaborative process modeling [26], where a group of people interact to construct a process model. The utilization of standard modeling languages may limit the differences between constructed models; however, different perspectives still exist, resulting in different models adhering semantics of the same language [24].

### 3.1 Case study

The LIFE ReWeee Initiative aims to prevent the creation of Waste Electrical and Electronic Equipment (WEEE) and is coordinated and employed by Appliances Recycling SA. This initiative includes a major action promoting the donation and exchange of EEE in a national-scale fashion, while a

web-based collaborative platform, namely the ReWeee platform, should be developed to bring together organizations and individuals participate in this action [27]. The main goal of the platform to facilitate and promote Electrical and Electronic Equipment exchange and donation among households or households and public/private bodies. Its success lies within the social communication between volunteers and their collaboration in order to achieve the best possible result.

We considered the EEE exchange process as a case to be completed with the collaboration of all interested parties. Our first task was to obtain an abstract description of the case from Appliances Recycling SA. When talking about a collaborative platform, our client had in mind a sort of social application or wiki, thus, the following paragraphs, in italics, is the first-level description of EEE exchange case as pro-

vided to us. When described by ReWeee management team, a user-centered approach was followed [31], based on the assumption that they were prescribing what different kinds of participants should do in the context of a collaboration environment.

*“First of all, there are two types of users. These are guest and registered users that differentiate themselves in the permissions that they get granted as far as the use of the platform is concerned.*

*More specifically, when any unregistered user visits the web platform for the first time, he gets prompted to register, by creating a user account. This account can be created either by signing up via an email and a password or by signing up via a social network account, which requires his/her giving to the platform the necessary permissions for using personal data. After a successful registration, the, from now on, registered platform user, is able to submit an advertisement donating or exchanging an item, to declare interest for an existing EEE product and propose an offer to acquire it, as well as to communicate with any other user who owns a desirable electric device.*

*Moreover, a registered user is not only able to search a product based on some conditions, namely, filters like item categories, item state, donating-user region, but also to either suggest changes regarding the item’s category for which he/she is searching, or even to comment in an advertisement that he/she had expressed interest for. That way, the appropriate users will be notified for either the category change proposal or the commenting in an advertisement.*

*Finally, registered users have a profile in which they are able to be notified for any recycling actions taken via a news-feed as well as being informed for general topics regarding recycling and its benefits. Within each user’s profile, a calendar exists via which a user can be informed for any recycling events taking place.”*

Based on the description provided, one could easily conclude that EEE exchange could not be effectively modeled using BPMN, as the sequence of activities unpredictable and random. However, a visual model of the process would be useful during user requirement analysis stage. For that reason, we identified it as a Case that could be modeled using CMMN, which enables the modeling of such activities in a more fluid fashion.

### 3.2 Case study lifecycle

The lifecycle of the EEE exchange modeling process is shown in Fig. 1. Two different modeling groups participate in the study.

As was also mentioned in [32], each group consist of a senior business process modeling expert, a junior mod-

eler and a senior software engineer. They are led by the senior modeling expert. Each group consists of two men and one woman. Senior modeling experts are over 40 years old, one man and one woman. The rest group members are aged 25–35. Both senior modeling experts were familiar with CMMN, while all modelers were familiar with collaborative processes and their requirements. All members were familiar with BPMN and related tools.

#### 3.2.1 Process analysis

**Requirements analysis** The whole process was initiated with an interview stage where the two working groups had a chance to independently interview ReWeee Initiative management team as well as other project stakeholders. Both of them asked for additional information on the project, performing a requirement analysis for the design of the collaborative process identified within the ReWeee web-based platform workflow.

**Modeling process** During the modeling procedure, there was no interaction between the two groups, as each one of them should create a model, using CMMN, for the analysis phase of the ReWeee project based on the information they collected during requirement analysis. They had the chance to re-consult with ReWeee Initiative management team members to clarify specific issues on EEE exchange process.

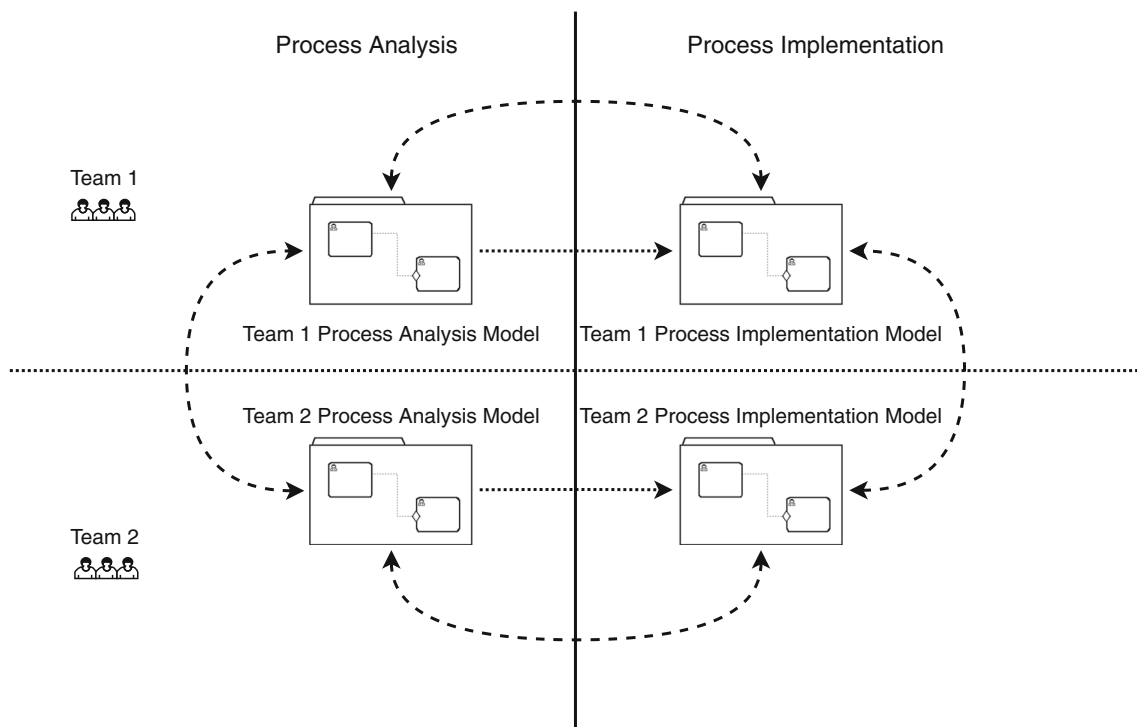
**Tools used** For the construction of their CMMN model, both modeling groups used Camunda Modeler tool, an open source version, as it provided full support for all of CMMN notation elements, being a cross-platform modeling tool. It is also quite easy to use.

**Recording modeling experience** After completing the construction of their models, each group’s experience with CMMN was recorded through a structured interview. Each group was provided with a structured questionnaire enabling them to record their views on the queries, shown in Table 2. These referred to specific aspects of their modeling experience for analysis purposes. The group members recorded their opinions as a group, though individual thoughts were also recorded. After filling the questionnaire, they were interviewed to better explain their views.

As mentioned in [32], the two models produced at the analysis phase of the EEE exchange process, despite the fact that they were valid in terms of utilizing CMMN concepts, seemed extensively different, which was not an anticipated outcome [32].

#### 3.2.2 Process implementation

The next phase of ReWEEE case study was the implementation of the process. We decided to explore the implementation



**Fig. 1** Empirical evaluation of EEE exchange process modeling using CMMN

of EEE exchange process using a CMMN execution tool. Having in mind, the aforementioned difference of the perspectives of the two models, we retained both the existence and the composition of the two groups, again as two alternatives for process implementation, aiming to the best possible result.

**Implementation Requirements** At that point, the two modeling groups interviewed potential users of the ReWeee collaborative platform, including several members of both local and international non-government organizations that already had expressed their interest in using the platform to be created. They analyzed the platform user's requirements in order to design a fully executable CMMN model for the collaborative process of EEE exchange.

**Modeling Process** As far as the construction of the implementation model is concerned, both groups were given the chance to alter their models as they moved from analysis to implementation phase to ensure they were executable upon a CMMN execution engine. Both teams achieved to create an implementation model.

**Tools used** Camunda Modeler was chosen for analysis purposes. However, it had limited capabilities for implementation purposes, since a significant amount of functionality prescribed by CMMN standard for CMMN execution engines has to be written by the engineers in Java code. After consultation with both groups, it was decided to both use Flowable suite for implementation purposes.

**Recording modeling experience** After completing the construction of their models, each group's experience with CMMN for implementation purposes was recorded through a structured interview. Each group was provided with a structured questionnaire enabling them to record their views on the queries, shown in Table 4. These referred to specific aspects of their modeling experience for implementation purposes. The group members recorded their opinions as a group, though individual thoughts were also recorded. After filling the questionnaire, they were interviewed to better explain their views. These referred to specific aspects of their modeling perspectives implementation such as the conformance with their modeling target. Additionally, their view on CMMN notation elements support provided by Flowable platform is also recorded in Table 3.

### 3.3 Model evaluation

The two modeling groups were interviewed after each phase upon several queries regarding their experience in order to identify the boundaries and limitations they came up with during their modeling with CMMN. To provide a comparative analysis, the models themselves should also be compared. To be able to evaluate the models as each group made the shift from the analysis to the implementation phase, as well as the models of the two groups in both phases, we have adopted existing CMMN model evaluation metrics regarding their complexity [19] and expressiveness [7]. The

evaluation results lead to observations regarding how modeling philosophy is altered during the shift from analysis to implementation. This analysis, combined with the experience of the process engineers, which is recorded through discussion, resulted in reflections on CMMN applicability for modeling collaborative processes in not only the analysis, but also the implementation phase.

## 4 Collaborative process analysis

Regarding the EEE exchange process analysis phase, the two models produced by the two modeling teams were extensively different, a not anticipated outcome, as discussed in the following. We consider both models to be valid in terms of utilizing CMMN concepts, as no syntax errors were observed. However, the two models project two different perspectives as far as both the level of analysis and design philosophy is concerned [32], as shown in Figs. 2 and 3.

### 4.1 Analytic perspective

The first modeling team created the model projected in Fig. 2. What can be commented for this modeling attempt is that its design logic was based in analytically depicting what the user of the collaborative platform should be enabled to do, as was recorded during requirement analysis. As the stakeholders had in mind a description of the case as a sequence of available screens by a collaboration platform, the modeling team adapted their perspective. What is primarily modeled is which are the events that lead the case from one state to another (in practice from one screen to another) without having in mind which activities are mandatory or not. For that reason, there is no use of discretionary tasks [22], while the majority of actions are linked to each other. Finally, data queried during the platform life-cycle are emphasized, while EEE exchange case milestones are not modeled. It is attempted to describe the case in detail so as to ensure that nothing is going to be skipped during implementation.

### 4.2 Abstractive perspective

The second modeling team designed, for the EEE exchange process, the model which is shown in Fig. 3. It projects, like the first one, the whole functionality provided from the ReWeee platform, but in contrast to the first one it represents it in an abstractive manner. It emphasizes which actions are mandatory for the EEE exchange case to be completed successfully. More specifically, there is an extensive use of discretionary tasks, while there is a large amount of activities that are not connected with each other. Data created during the case are hardly under consideration in this model, but on the contrary milestones are defined to highlight anything that

is considered as important for the ReWeee case. Stages are also modeled in order to isolate the less important parts. What could be commented is that this working group attempted to ensure that the end-user has understood the EEE exchange case components.

### 4.3 Rising questions

As the first model (see Fig. 2) seems to be more complex and sequence-oriented than the second one (see Fig. 3), one could comment that it looks in way quite alike as a BPMN model, in terms of identification of flow.

In the following, we discuss the results of interviewing both teams and explore what drove them to adopt their perspective. The questions set to two modeling groups are presented, alongside with the arguments they gave. The modeling experience of both groups was discussed, with some queries as a basis, in order to explore their willingness to adopt CMMN in modeling similar cases. These queries referred to: the modeling objective of their attempt, what was facilitating during the modeling procedure, what difficulties they faced in modeling the case and what would be the difference in modeling the EEE exchange case with BPMN. Table 2 presents a summary of the arguments provided from the modeling groups.

In principal, the analytic model (Fig. 2) had as its primary aim to describe exchange case flow in detail, having the visitor of ReWeee platform in mind. It was an effort to precisely describe the user experience expected as exposed by ReWeee management team. Emphasis was given to the correlation of tasks representing each possible step of the overall process in great detail. Furthermore, focus is given to depict the way EEE exchange case will be automated, without missing any of the projected functionality. That leads to a quite descriptive model. On the other hand, the abstractive model (Fig. 3), attempts to define which ones of the case activities are mandatory so as to complete the case by making extensive use of the Discretionary Task notation element. That way, a high level of abstraction is provided to the model, making it easily readable. Though it lacks in information about case data or how model elements are correlated.

Moreover, the ease of modeling was also discussed with the two modeling groups. For the one that modeled EEE exchange case in an analytic way, Sentries [22] were the most facilitating feature of CMMN, enabling them to model transitions in the case's state. Having the projection of the sequence of activities as their main goal, this feature was the one that helped them the most at their modeling attempt. The other group, that created the abstractive model, find the ease of modeling lying to the use of Discretionary Tasks for modeling less important Tasks. That way, the mandatory Tasks, that lead to case completion, could be highlighted.

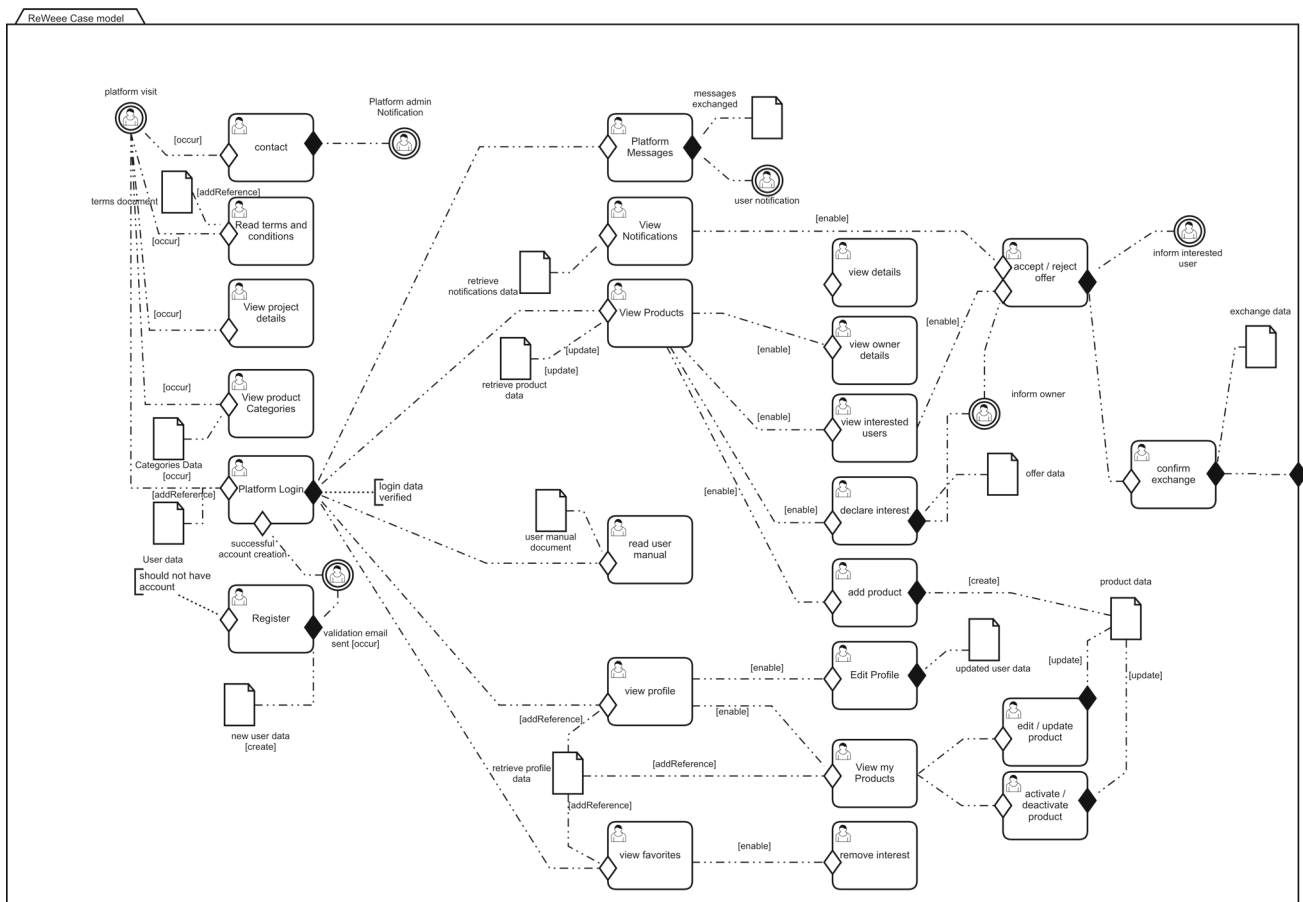


Fig. 2 EEE exchange case from an analytic perspective

On the other hand, both of the groups referred facing difficulties during the modeling of the EEE exchange case. For the first group, that modeled the process analytically, it was difficult to identify how to model the flow of the process. Namely, they could not identify easily how sequence is represented. For that reason they utilized extensively the notion of Sentry, using events not only to project the control of flow but also to describe how the completion of a Task could commence another Task. However, that made their model quite complex, not familiar with CMMN. On the contrary, for the other group that modeled the EEE exchange process in a more abstractive fashion, it was difficult to define the exact correlation between the case activities. That led them to model a mass of Tasks, independent to each other, which was quite unusual for the modelers and that also led to an undefined sequence of activities.

Finally, concerning the differentiation from BPMN, the analytic model had a design philosophy very close to the one of the BPM, as it had the representation of the sequence of the case activities as its main objective. Extensive data modeling and strict sequence definition made the modeling very close to the philosophy of BPM. The

final model looked different from a corresponding BPMN model just because the differentiation in notation elements. On the contrary, the abstractive model projected the functionality provided by the projection of the Tasks available but in a quite different way from a BPM model. There was not such a thing like sequence representation, while data were completely out of frame. This fact led the model to look like more than a definition of Tasks without taking under consideration the correlation between them.

## 5 Collaborative process implementation

### 5.1 Implementation support in CMMN-enabled tools

The next step for the two working groups was to create the implementation model for the EEE exchange process. In order to do so, they should enhance their models to become executable using a CMMN execution engine.

Camunda BPM Platform was the open source platform for workflow and decision automation, used during the analysis



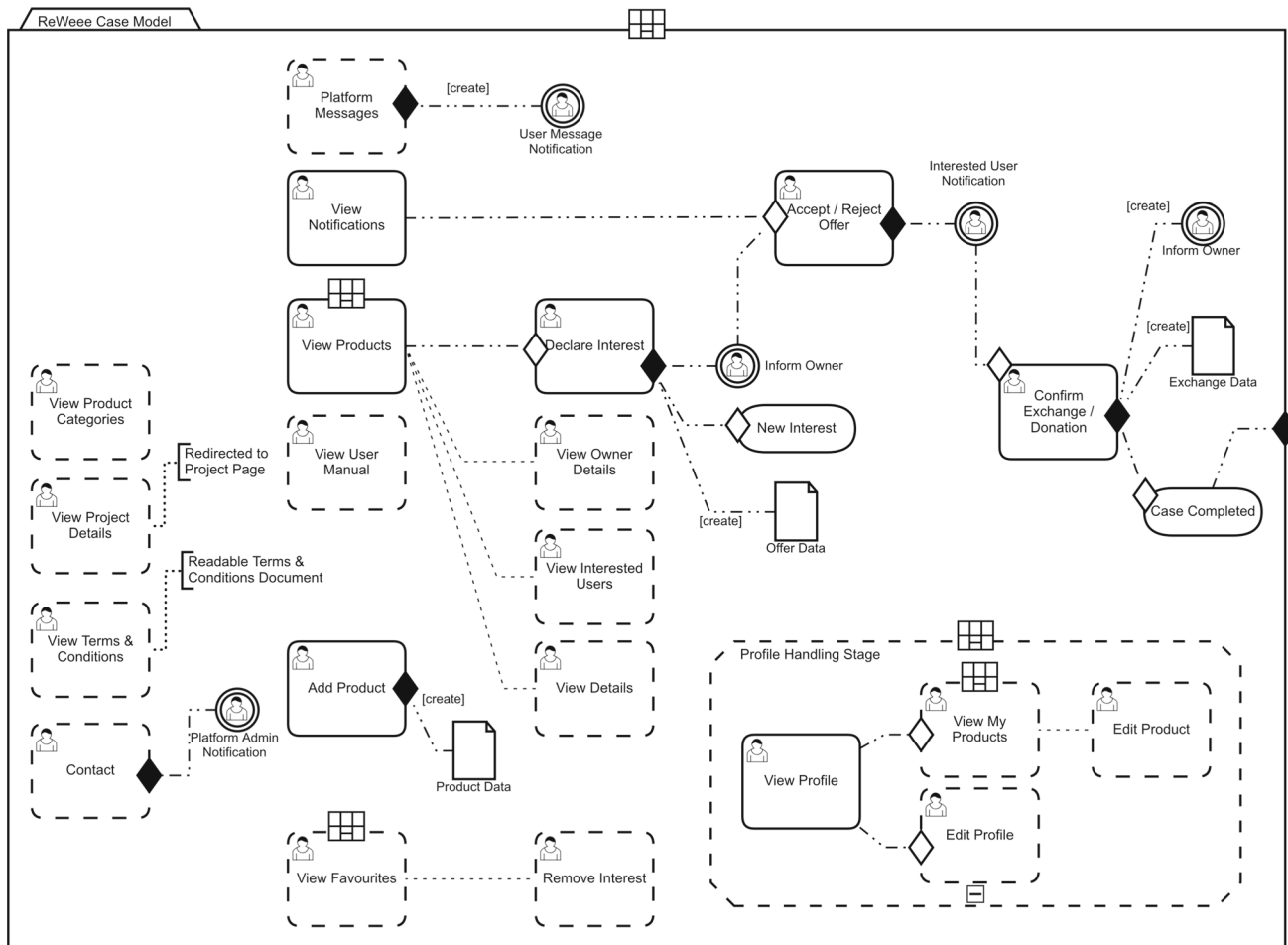


Fig. 3 EEE exchange case from an abstractive perspective

Table 2 Queries and arguments for the two modeling perspectives

Queries	Analytic perspective	Abstractive perspective
Modeling target	Design the EEE exchange case, as described by end-users, representing tasks in detail so as to ensure optimal implementation	Define the mandatory tasks of the case to highlight those that when instantiated could lead to case completion
Ease of modeling	Modeling was mainly facilitated by the use of Sentries. It helped modelers achieve the objective of modeling the sequence of activities	Modeling was mainly facilitated by the use of Discretionary Tasks. Easier highlighting of the mandatory Tasks for the case completion
Modeling difficulties	Difficulties in understanding how to represent the flow of the case and to comply with the philosophy of CMMN standard.	Difficulties in defining the correlation between the Tasks as well as in identifying how each task can be instantiated
BPMN differentiation	Design philosophy was very close to the one of BPMN as the sequence of Tasks was the main objective of modeling	Platform functionality and Tasks were modeled according to the CMMN standard in a quite different way from BPMN

phase. However, it had limited capabilities for implementation purposes, since a significant amount of functionality prescribed by CMMN standard for CMMN execution engines has to be written by the engineers in Java code. An alternative platform with a CMMN engine conforming

closer with CMMN specification was chosen, named Flowable.

As the adoption of CMMN standard is in its initial stage, CMMN execution engines do not fully comply to CMMN standard yet. Table 3 projects the main CMMN elements

**Table 3** Executable models elements tool support

Model elements	Tool usage	Tool support
Case Plan Item	It is handled as a container for all other CMMN elements. Case Plan Model is generally supported by CMMN-enabled tools	●
Human Task Plan Item	The implementation of human tasks includes the creation of forms to ease their completion and their assignment to specific users. Human tasks are completely supported	●
Case File Item	During design-time case file items can be modeled by case designers; however, during run-time, they are not utilized during run-time. Data should be attached into forms and are handled through them	○
Event Listener Plan Item	Events are included in the models, connected with the sentries these trigger. However, during run-time their implementation is usually unclear and tricky	○
Sentries	Entry and exit criteria are modeled during design-time, attached upon the tasks they control. However, their implementation is buggy or requires hard-coding	○
Stage Plan Item	Stage plan items like human tasks are generally supported by CMMN-enabled tools. Their implementation is a straightforward task	●
Milestone Plan Item	Milestones are easily designed during design-time enabling goal-oriented modeling for cases. However, their implementation is quite unclear as it is recorded only in case logs and not visualized	○
Discretionary Items	Discretionary Items are modeled in CMMN executable models. However, these are not currently supported at run-time environment	○

that were used during modeling, in addition to how these elements can be executed in CMMN execution engines. The level of support is projected as follows: supported (black dot), partly supported (gray dot), and not supported (white dot).

Flowable fully supports CMMN notation elements, while it provides adequate means for their execution, although some of them are not fully supported yet, as shown in Table 3.

## 5.2 Implementation of perspectives

During the implementation phase, initial models should be enriched with necessary information to ensure their executability. In this process, the two groups were instructed to neglect the lack of full support of specific features of CMMN in Flowable, such as the buggy implementation of Sentries. They were prompted to concentrate in CMMN execution features, as prescribed in the standard, and consult Flowable future improvements release notes.

The models created were different between them; however, they were also different from their initial versions for analysis purposes. The models for the implementation phase are projected in Figs. 4 and 5.

### 5.2.1 Analytic perspective

The implementation model of the first modeling group, which had modeled its design-time model with an analytic perspective, required alterations. As it is described in Table 3, data objects could not be model during run-time. Thus, for the execution of their model, what was required, was for the first modeling group, the removal of data objects as CMMN model elements. However, one could consider the exclusion of data objects in the model, alongside with the sentries that controlled these data objects, as the only difference between the models shown in Figs. 2 and 4.

### 5.2.2 Abstractive perspective

The second modeling group, which modeled with an abstractive perspective, during the implementation phase of the EEE exchange process modeling procedure, made changes to their previous model. As it is shown in Fig. 5, just like for the analytic one, data objects could not be included in the model. Thus, case file items were excluded. In addition, what also could not be modeled during run-time was discretionary items. For that reason, the designers of the second modeling group decided to include all the available options for process

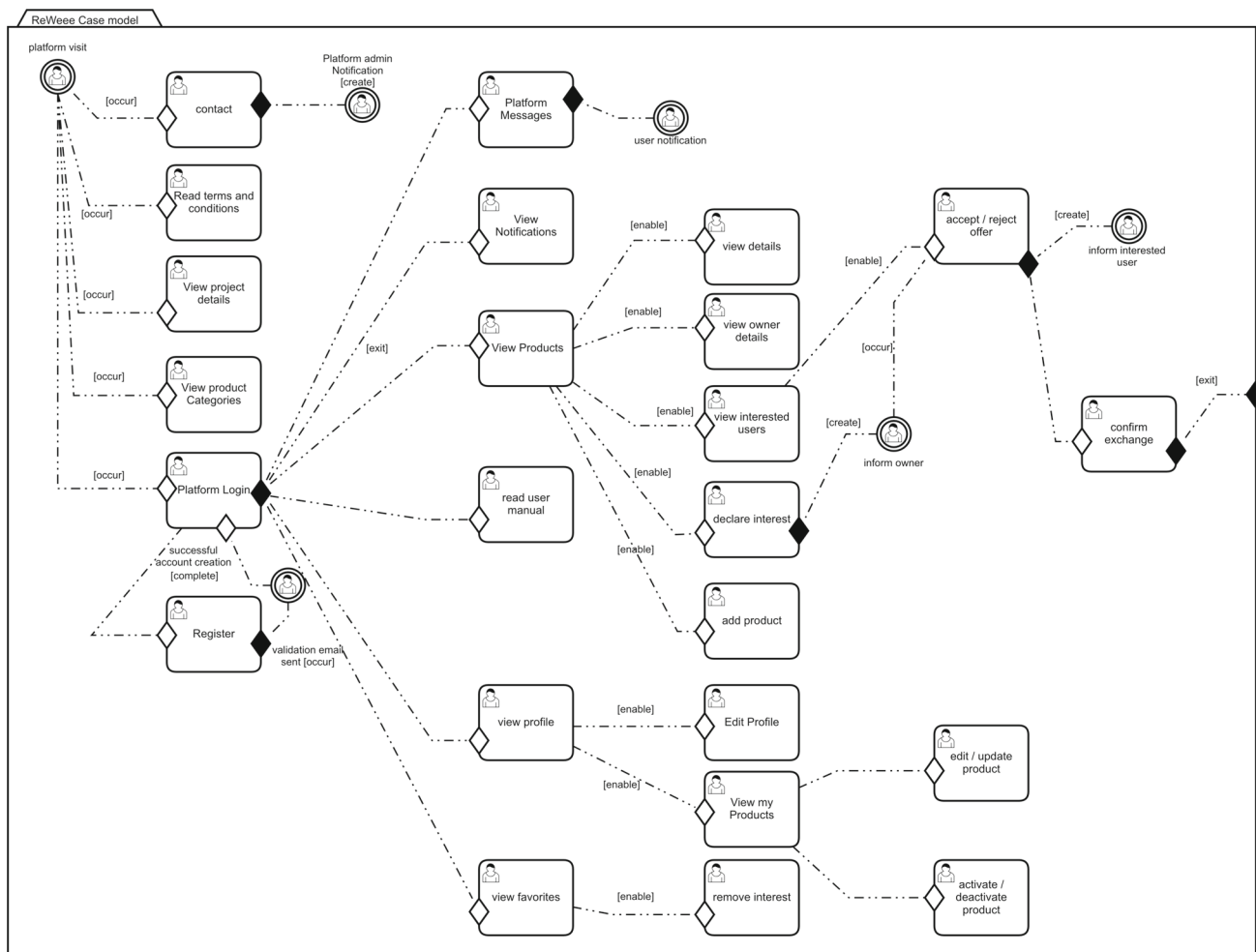


Fig. 4 EEE exchange case from an analytic perspective

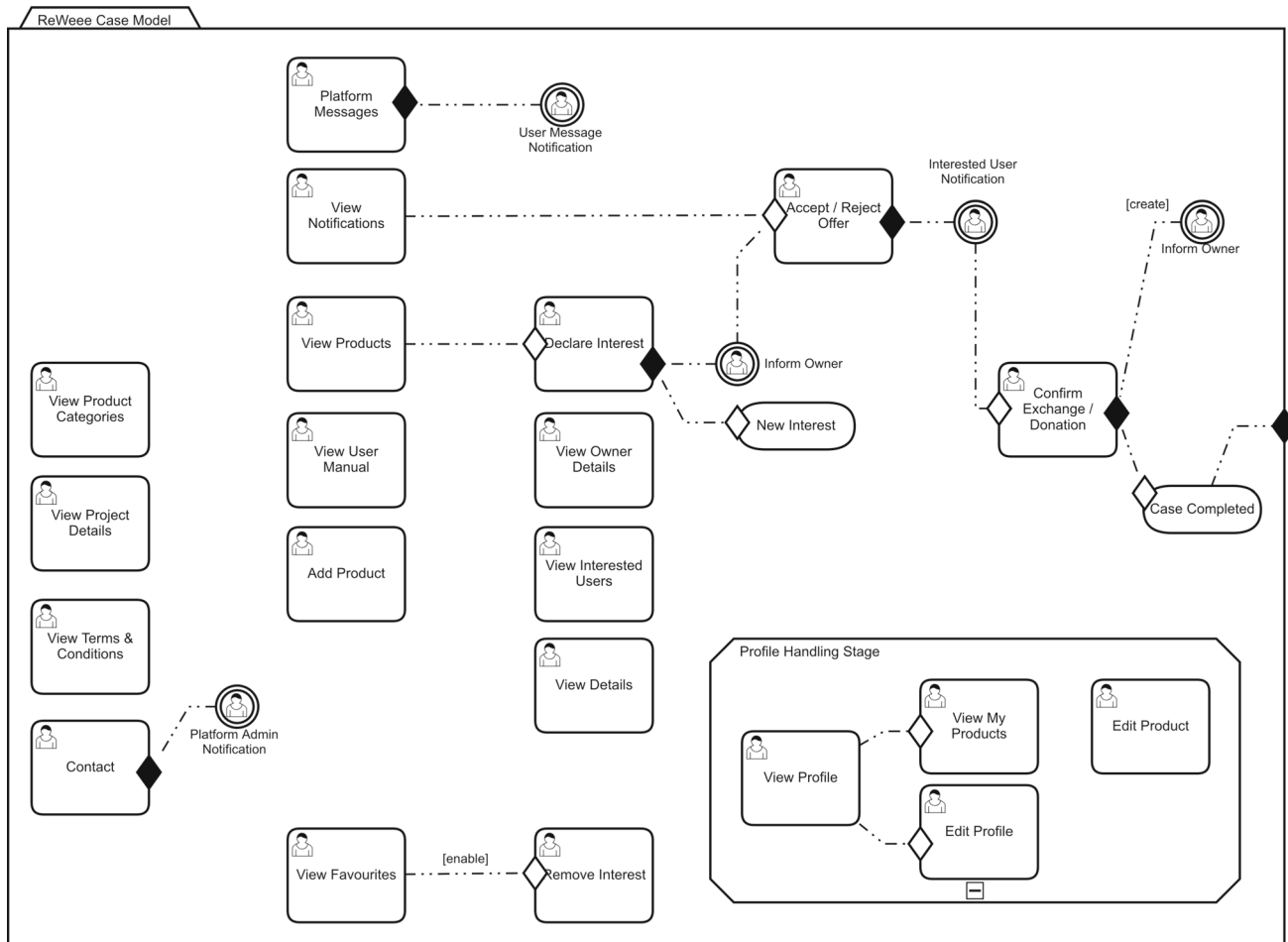
activities during process implementation, as normal human tasks available to the involved process actors.

### 5.3 Rising questions

After the presentation of their implementation models, the two modeling groups have been interviewed. The experience of both groups was discussed, in order to identify what they found challenging during the implementation of their perspectives on the collaborative process, as well as how challenging it is to use CMMN-enabled tools in order to implement their models for the collaborative process. The queries set to the two modeling groups are presented, alongside with the arguments given from them. These queries referred to: how sequence is represented, how data flow is handled, whether there is flexibility in run-time, whether there could be decision-, event- or goal-driven implemen-

tation for the perspectives, how case in both perspectives could be completed and whether the implementation conformed with the modeling target of each perspective. The following table (Table 4) presents a summary of the arguments provided from the modeling groups.

To begin with, as far as the representation of sequence is concerned, for the analytic perspective, despite the fact that there were many tasks to the model, available to be implemented by the case workers, the basic path to case completion was defined in design-time. However, the slight support on sentries made difficult the representation of sequence. On the other hand, due to the philosophy of the abstractive perspective, sequence was scarcely implemented in the first place. However, the main path to case completion was predefined in design-time. The limitation, similarly to the analytic perspective, was the slight support of sentries.



**Fig. 5** EEE exchange case from an abstractive perspective

Considering data flow management, the first modeling group found difficulties in representing the flow of data in run-time. For them, the data that each task was creating / updating should be included in the form that was created to visualize this task. On the contrary, for the second group, the fact that there was scarce data modeling in the design-time, hid the inability of data modeling in run-time, as only a few data entities should be included in forms.

Regarding the flexibility in run-time that could be provided to the case workers from each perspective, for the analytic perspective, the model was already rigid at design-time, leaving the impression that every task on the model should be executed. There was no difference on run-time, as the executable model was not including any alterations to design philosophy. For the abstractive one, due to exclusion of discretionary tasks, the flexibility, provided in design-time through their extensive use, was reduced during implementation, as every discretionary task was transformed into a plain task.

Moreover, the discussion focused on the capability of applying different aspects and styles of implementation,

namely decision-, event-, and goal-based implementation. For the analytic perspective, the slight capability of implementing sentries, that were used in order to model decisions upon the case flow, and events, tackled the implementation of decisions nodes and actions driven-by-events in run-time. Additionally, the philosophy of the analytic perspective was not aligned with goal-oriented modeling, as milestones were absent from design-time and there was no need of implementing them. On the other hand, for the abstractive perspective, despite the fact that there was a scarce use of sentries in the design-time, which camouflaged the issue with the slight capability of implementing decisions, the difficult implementation of event listeners tackled the adoption of an event-driven logic for the case. On the contrary, milestones were retained in the model for the second group (Fig. 5) as their implementation was possible even if that was recorded only in logs.

Respecting the  $\langle \rangle$  condition that could lead in case completion, that could be possible through the completion of the exit criterion attached upon the Case Plan Item. This condition was the same on both perspectives, and despite the slight

**Table 4** Queries and arguments for the two modeling perspectives implementation

Queries	Analytic perspective	Abstractive perspective
Representation of sequence	The basic path to case completion was defined from the analysis phase. The representation of sequence was performed through the use of sentries	Sequence was scarcely depicted in the models, as a sequence was only defined on the path for case completion
Data handling	Difficulties in representing data handling on the implementation model.	Data should be included in forms generated for the completion of tasks
Flexibility in run-time	The model was already rigid from the analysis phase. No alterations to design philosophy was need	Flexibility was reduced in the implementation model. Every discretionary task was transformed to normal human task
Decision-based implementation	The fact that discretionary tasks could not be implemented in run-time through different sequence paths	the implementation of decision logic. Decisions could only be represented by
Event-driven implementation	The implementation of events was tricky and unclear, thus the implementation of an event-driven logic for the case was difficult. Sequence driven by user notification and interaction (user events) was implemented	
Goal-oriented modeling	Modeling philosophy could not be considered goal-oriented. Milestones were absent from both design and implementation	Milestones continued to be present in the model. Their implementation was possible even if that was recorded only in logs
Case completion	Case completion required the completion of the case exit criterion. The completion of the previous task or the achievement of the previous milestone	The entry for case completion was easily implemented. It was triggered by either the
Modeling target conformance	The modeling target was to ensure optimal implementation. The analytic perspective was implemented, although it cannot be considered optimal	The modeling target was to define the mandatory tasks for case completion. Implementation is conformed with its modeling target
Future improvement	Case file elements are useless at run-time. Requires improvement on executable language notation for data management which is considered crucial for implementation	More suitable for flexible processes like EEF exchange process. CMMN tools should provide greater flexibility at run-time

capability of implementing sentries, the sentry for case completion was easily implemented as it was triggered by the time the previous task (analytic perspective) or milestone (abstract perspective) was successfully completed or achieved.

Another point of discussion questioned the conformance with the modeling target of each perspective. The modeling target of the first group was to design the EEE exchange process in detail so as to ensure optimal implementation (see Table 2). Their analytic perspective was implemented despite the platform's limitations, although it can not be considered optimal. On the other hand, the modeling target for the second group was to define the mandatory tasks that could lead to the case completion (see Table 2). From the implementation model projected in Fig. 5, one could assume that it is conformed with its modeling target.

## 6 Evaluation

In order to evaluate the applicability of CMMN for collaborative processes modeling not only in analysis but also in the implementation phase, one should explore the following questions [28,33]:

- Q1: What were the boundaries regarding CMMN on modeling of a collaborative process?
- Q2: Could the resulting models be executable? What was the effort needed to implement the process?
- Q3: Was the support provided by CMMN-enabled platforms sufficient or there were limitations on their use for both the design and the implementation phase?
- Q4: What kind of improvements could be proposed for CMMN?

The procedure of process modeling ended with some fruitful reflections regarding how the two groups created their models. As aforementioned, they came up with models that were based upon two different perspectives, having, according to their mentions, different modeling targets. More specifically, the first modeling group (Fig. 2) seemed to focus on modeling the automation of EEE exchange process, handling tasks in a unified fashion, while the second (Fig. 3) focused exactly on the opposite, namely to identify the main tasks and make a distinction between mandatory and optional tasks. Upon implementation, this differentiation on modeling perspectives was retained, creating two also different models (Figs. 4 and 5).

### 6.1 Models evaluation

In order to evaluate the differentiation in perspectives, evaluation metrics for the models were reviewed. Firstly, we decided to measure the complexity of the models, as it is

**Table 5** CMMN elements weights as appear in [19]

CMMN element	Weight
Case	1
Stage	1
Discretionary stage	2
Plan fragment	3
Case file item	1
Task	1
Discretionary task	2
Event listener	2
Milestone	1
Connector(sentry)	0
Entry criterion with connector	1
Exit criterion with connector	1

**Table 6** Complexity results for team 1 models per phase

Metrics	Analysis	Implementation
Size	102	62
Length	2	2
Complexity	92	70

**Table 7** Complexity results for team 2 models per phase

Metrics	Analysis	Implementation
Size	56	44
Length	3	3
Complexity	67	49

described in [19], where three complexity metrics are being introduced, namely size, length and complexity. In [19], the size of a model is defined as the sum of the CMMN elements that exist in a model. These elements can be a selection of cases, stages and discretionary stages, fragments, case file items, tasks and discretionary tasks, event listeners, milestones and connectors. Length is defined as the maximum nested depth of a CMMN model. Finally, complexity is defined as the sum of the weights, that each element has. Weights are defined in [19], and are projected in Table 5.

Thus, for the models that were created by the two working groups, we measured size, length and complexity, as evaluation metrics for the model complexity. For example, the size of the model 2 is: Size = Cases + Stages + Tasks + Case File items + Event Listeners + Milestones + Connectors = 102. As far as the model length is concerned, we have only a case element containing tasks, namely two levels, thus length = 2. For complexity, we sum weights of contained elements according to Table 5 and the result equals 92. To study more easily the measurements, we grouped them per perspective. Tables 6

**Table 8** Expressiveness results for team 1 models per phase

Patterns	Analysis	Implementation
Control flow	19	21
Merge and split	5	5
Advanced branching	0	1
Exit criteria	13	8
Data	15	0
User interference	7	7
Milestones	0	0

**Table 9** Expressiveness results for team 2 models per phase

Patterns	Analysis	Implementation
Control flow	6	7
Merge and split	3	2
Advanced branching	0	0
Exit criteria	7	6
Data	0	0
User interference	2	2
Milestones	2	2

and 7 project the separate measurements for design and implementation.

Secondly, we decided to study another evaluation metric, Expressiveness. In [7], workflow patterns, that describe the expressiveness of a CMMN model, are being recognized, as metrics for evaluating CMMN models. Workflow patterns introduced in [39] and extended [34–36] describe process modeling requirements in an implementation independent manner and distill the essential features of business process modeling languages [39]. Several of the patterns introduced in [7] were chosen for our evaluation, describing different aspects of the models. In our case, we intended to measure expressiveness of the models quantitatively. For that reason, for each pattern, we have summed up how many times this pattern could be identified in a model. The greater the sum, the greater the expressiveness of the model. Tables 8 and 9, present the results each group.

## 6.2 Observations

### *Comments on Complexity*

1. The first working group (analytic perspective), created an analysis model, quite large in size and quite complex in logic (see Table 6). Due to their attempt to describe everything that one would like to know for the case before implementing it, a large amount of CMMN elements was used, especially for the representation of data-task correlations.

2. The model for the analytic perspective has been converted to a much less complex model during the shift from analysis to implementation. The majority of complexity metrics were decreased significantly. The reason behind this decrease lies in the exclusion of case files from the implementation model. Data management is implemented complementary to CMMN notation.
3. The second working group adopted an abstractive perspective to design the case. Thus, their model was much simpler, smaller in size, less complex but with greater depth (larger length), than the one created by the first group (see Table 7). The main modifications during implementation concentrated at replacing discretionary elements with simple ones, restricting expressiveness and consequently size and complexity of the model.

### *Comments on Expressiveness*

1. The model created, based on the analytic perspective of the first working group, was quite complex and had a high level of expressiveness. As shown in Table 8, many workflow patterns were identified, leading to the assumption that a complex model at least can be expressive. Besides, that was also the modeling target of the first working group.
2. Despite the fact that the complexity metrics for the analytic model were decreased, it retained a large amount of its expressiveness, through patterns that could be supported during implementation, as for example control flow patterns (see Table 8). Data patterns were eliminated from the implementation model, and were deemed useless at run-time.
3. On the contrary, the abstractive perspective model of the second working group retained its expressiveness in the transition from the analysis to the implementation model. Very few patterns could be identified, mainly due to lack of both sequence between the various tasks and data handling representation.

### *General Observations*

1. The models representing the analytic perspective were complex, expressive, in terms of patterns, and large in size containing information for nearly everything required for the implementation of the case. These models were influenced by the CMMN execution semantics, as some of their key characteristics, such as data modeling, have been limited in practice upon implementation. However, their expressiveness was not limited, as the majority of their data-driven flow was embedded into the control flow. That required the representation of data objects through the forms that each task was creating at run-time. Namely, data were represented as prerequisites for

tasks completion through the use of forms, as prescribed by CMMN standard anyway.

On the other hand, the models representing the abstractive perspective were much simpler, with slight description of flow and not as complex as the ones of the analytic perspective. As the analysis model did not include case file elements, no such alteration was necessary. They acted as if they were aware of CMMN limitations in data management. As they included less tasks, constructing related forms during implementation was a more complex work, as well as related data management. However, their main simplification was the elimination of discretionary events. This has nothing to do with CMMN semantics, but is related to the current features of CMMN engines. This was a very helpful feature they had to abolish.

2. Another general observation one could make is that the differentiation on perspectives was sustained from analysis to implementation phase, which was not an expected outcome, a fact that was also reflected in the evaluation stage. More specifically, despite the alterations that the models underwent during the transition from the analysis to implementation, both implementation models were completely different as far as both complexity and expressiveness are concerned. At this stage, one should consider the fact that both groups were valid as far as the use of CMMN is concerned, at every phase of the modeling procedure. Additionally, any alterations that were made by the working groups during implementation did not affect neither the modeling philosophy of each group or the nature of the case study.

### 6.3 Reflections on evaluation questions

In order to complete the exploration of CMMN applicability regarding the modeling of collaborative processes, one should attempt to answer the evaluation questions set in the beginning of this section.

- Q1: *What were the boundaries regarding CMMN on modeling of a collaborative process?*
- A1: CMMN notation and semantics covered the modeling requirements of both working groups in an above acceptable level. The two different modeling perspectives were clearly projected, representing two different modeling targets as these are described in Table 2.
- Q2: *Could the resulting models be executable? What was the effort needed to implement the process?*
- A2: During the implementation of their perspectives, the two working groups had to make alterations to their initial models due to the fact that few of the used notation, are not yet fully supported by CMMN execution engines (Table 3). However, despite the boundaries they came up

with, both working groups managed to create executable models using the available CMMN semantics provided in [22]. Additional programming effort was needed by both groups to implement forms and data structures during implementation. They also had to add code to manage entries and event implementation (see Table 4).

- Q3: *Was the support provided by CMMN-enabled platforms sufficient or there were limitations on their use for both the analysis and the implementation phase?*

A3: CMMN notation support was adequate by all tools; however, the features of CMMN execution engines as prescribed in the standard are not fully supported yet. The lack of effective execution support of important notation elements, such as sentries and discretionary items, limits CMMN applicability.

- Q4: *What kind of improvements could be proposed for CMMN? For CMMN tools?*

A4: As it is shown in Table 4, the two working groups were questioned regarding further improvement of CMMN. Both of them, proposed improvements, with the first working group to confirm that CMMN requires improvement on semantics for case files, in order to become useful for implementation purposes by CMMN-tools. The second group, confirmed that CMMN is more appropriate for modeling flexible processes than BPMN-like languages. However, they anticipated better support by CMMN engines at run-time.

### 6.4 Discussion

Regarding the results of CMMN application, the models created were valid in terms of language usage as no syntax error was found. Moreover, all CMMN notation elements were properly used and the models created covered the modeling requirements of the project, creating two different implementations for the EEE exchange platform's workflow. The aforementioned differentiation gave us the flexibility of choosing the most appropriate implementation, according to the project stakeholders' and potential users' needs. Despite the fact that CMMN successfully covered the modeling requirements of the real-world project that should be analysed, the two working groups faced challenges when it came to implement their models, forcing them to reconsider their modeling perspectives. The challenges concentrated on the weak support of new elements introduced in CMMN by corresponding execution engines at this point in time. As the implementation of these platform progresses, it is expected to fully support CMMN standard in the near future. However, vendors should have in mind that CMMN engines should operate in a more flexible fashion than BPMN engines to support even-driven workflow, necessary to take full advantage of useful notation elements, such as Discretionary Items



and Sentries, that provide modelers with flexibility in modeling.

Regarding CMMN language, another improvement could be the provision of more detailed semantics for data representation, in order to enhance data-driven implementation of collaborative processes, which was another major issue that both working groups faced when they were asked to implement their models.

Nevertheless, the fact that, despite any boundaries, CMMN could be used for both the analysis and the implementation of a flexible process like EEE exchange process, is very promising, detaching CMMN from the sequential logic of BPMN-like languages.

## 7 Conclusions and future work

There are recent works presenting the application of CMMN, alongside with other modeling languages, like BPMN and DMN, to model specific case studies [41]. The research that underwent in this paper is the first that evaluates the application of CMMN on a real-world project, based on the project participants' experience. What can be concluded is that we have a new modeling language that can cover the modeling requirements for flexible and variable processes, such as collaborative processes, in both their analysis and implementation phases.

In the future, what we plan to examine more thoroughly, is the willingness to adopt CMMN, for modeling human-centric processes, of not only familiar-with-BPMN modelers but also of inexperienced users, considering the difficulties of learning a new non-traditional BPMN alternative as is mentioned in [16]. Additionally, advanced execution requirements for CMMN engines, implemented by CMMN-enabled tools, should be defined, in order to support enhance data representation and discretionary tasks upon implementation phase. Considering that Case Management as a theory is data-centric and decision-based, it is obvious why these two features should be supported.

**Acknowledgements** This work is partly co-funded by the European Commission through the LIFE+ Funding program, LIF14 ENV/GR/000858: LIFE REWEEE Development and demonstration of Waste Electric and Electronic Equipment (WEEE) prevention and reuse paradigm.

## References

- Alexopoulou, N., Nikolaidou, M., Kanellis, P., Mantzana, V., Anagnostopoulos, D., Martakos, D.: Infusing agility in business processes through an event-centric approach. *Int. J. Bus. Inf. Syst.* **6**(1), 58–78 (2010). <https://doi.org/10.1504/IJBIS.2010.034005>
- Antunes, P., Herskovic, V., Ochoa, S.F., Pino, J.A.: Modeling highly collaborative processes. In: Proceedings of the 2013 IEEE 17th International Conference on Computer Supported Cooperative Work in Design (CSCWD), pp. 184–189 (2013)
- Azadegan, A., Papamichail, K.N., Sampaio, P.: Applying collaborative process design to user requirements elicitation: a case study. *Comput. Ind.* **64**(7), 798–812 (2013)
- Birger Lantow, J.W., Herzog, P.: Method and notation application for case model creation in the social sector. *Complex Syst. Inform. Model. Q.* **13**(123), 36–50 (2019). <https://doi.org/10.7250/csimq.2019-21.03>
- Bögel, S., Stieglitz, S., Meske, C.: A role model-based approach for modelling collaborative processes. *Bus. Process. Manag. J.* **20**(4), 598–614 (2014)
- Cai, J., Lu, S.C., Grobler, F., Case, M., Jing, N.: Modeling and managing collaborative processes over the internet. *Bus. Process. Manag. J.* **11**(3), 255–274 (2005)
- de Carvalho, R.M., Mili, H., Boubaker, A., Gonzalez-Huerta, J., Ringuette, S.: On the analysis of CMMN expressiveness: revisiting workflow patterns. In: 2016 IEEE 20th International Enterprise Distributed Object Computing Workshop (EDOCW), pp. 1–8. IEEE (2016)
- Ding, T., Gao, Y., Ma, Y.: A semantic preserving CEP based approach for business data processing. In: 2016 12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), pp. 1808–1815 (2016)
- Dustdar, S.: Caramba process-aware collaboration system supporting ad hoc and collaborative processes in virtual teams. *Distrib. Parallel Databases* **15**(1), 45–66 (2004)
- Grudzińska-Kuna, A.: Supporting knowledge workers: case management model and notation (CMMN). *Inf. Syst. Manag.* **2**, 3–11 (2013)
- Gruhn, V., Laue, R.: Adopting the cognitive complexity measure for business process models. In: 5th IEEE International Conference on Cognitive Informatics, 2006. ICCI 2006, vol. 1, pp. 236–241. IEEE (2006)
- Gruhn, V., Laue, R.: Complexity metrics for business process models. In: 9th International Conference on Business Information Systems (BIS 2006), Citeseer, vol. 85, pp. 1–12 (2006)
- Herzberg, N., Kirchner, K., Weske, M.: Modeling and monitoring variability in hospital treatments: a scenario using CMMN. In: International Conference on Business Process Management, Springer, pp. 3–15 (2014)
- Hewelt, M., Wolff, F., Mandal, S., Pufahl, L., Weske, M.: Towards a methodology for case model elicitation. In: Gulden, J., Reinhartz-Berger, I., Schmidt, R., Guerreiro, S., Guédria, W., Bera, P. (eds.) Enterprise, Business-Process and Information Systems Modeling, pp. 181–195. Springer, Berlin (2018)
- Kluza, K.: Measuring complexity of business process models integrated with rules. In: International Conference on Artificial Intelligence and Soft Computing, pp. 649–659. Springer (2015)
- Koutsopoulos, G., Bider, I.: Teaching and learning state-oriented business process modeling. Experience report. In: Reinhartz-Berger, I., Gulden, J., Nurcan, S., Guédria, W., Bera, P. (eds.) Enterprise, Business-Process and Information Systems Modeling, pp. 171–185. Springer, Cham (2017)
- Kurz, M., Schmidt, W., Fleischmann, A., Lederer, M.: Leveraging CMMN for ACM: examining the applicability of a new OMG standard for adaptive case management. In: Proceedings of the 7th International Conference on Subject-Oriented Business Process Management, p. 4. ACM (2015)
- Marin, M.A., Hauder, M., Matthes, F.: Case management: an evaluation of existing approaches for knowledge-intensive processes. In: International Conference on Business Process Management, pp. 5–16. Springer (2015)
- 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, p. 28. ACM (2015)
20. Motahari-Nezhad, H.R., Swenson, K.D.: Adaptive case management: overview and research challenges. In: 2013 IEEE 15th Conference on Business Informatics, pp. 264–269 (2013). <https://doi.org/10.1109/CBI.2013.44>
  21. Object Management Group: Decision model and notation. <http://www.omg.org/spec/DMN/> (2015). Accessed 1 Nov 2018
  22. Object Management Group: Case management model and notation v1.1.1. <http://www.omg.org/spec/CMMN/1.1/CMMN> (2016). Accessed 1 Nov 2018
  23. Plebani, P., Marrella, A., Mecella, M., Mizmizi, M., Pernici, B.: Multi-party business process resilience by-design: a data-centric perspective. In: International Conference on Advanced Information Systems Engineering, pp. 110–124. Springer (2017)
  24. Prilla, M., Nolte, A., Herrmann, T., Kolfshoten, G., Lukosch, S.: Collaborative usage and development of models: state of the art, challenges and opportunities. *Int. J. e-Collab.* **9**(4), 1–16 (2013)
  25. Rajsiri, V., Lorré, J.P., Bénaben, F., Pingaud, H.: Collaborative process definition using an ontology-based approach. In: Camarinha-Matos, L.M., Picard, W. (eds.) *Pervasive Collaborative Networks*, pp. 205–212. Springer, Boston (2008)
  26. Renger, M., Kolfshoten, G.L., De Vreede, G.J.: Challenges in collaborative modelling: a literature review and research agenda. *Int. J. Simul. Process Modell.* **4**(3–4), 248–263 (2008)
  27. ReWEEE Project: ReWeee Project basic information. <https://www.reweee.gr/en/basic-information.html> (2018). Accessed 22 Feb 2018
  28. Rolan, E., Chavira, G., Orozco, J., Soto, J.P.: Towards a framework for evaluating usability of business process models with BPMN in health sector. *Procedia Manuf.* **3**, 5603–5610 (2015). <https://doi.org/10.1016/j.promfg.2015.07.748>
  29. Rolón E, García, F., Ruiz, F., Piattini, M., Visaggio, C., Canfora, G.: Evaluation of BPMN models quality—a family of experiments. In: ENASE 2008—Proceedings of the 3rd International Conference on Evaluation of Novel Approaches to Software Engineering, Funchal, Madeira, Portugal, May 4–7, 2008 (2008)
  30. Routis, I., Stratigaki, C., Nikolaidou, M.: Exploring ACM and S-BPM for modelling human-centric processes: an empirical comparison. In: Proceedings of the 8th International Conference on Subject-Oriented Business Process Management, p. 5. ACM (2016)
  31. Routis, I., Nikolaidou, M., Anagnostopoulos, D.: Using CMMN to model social processes. In: International Conference on Business Process Management, pp. 335–347. Springer (2017)
  32. Routis, I., Nikolaidou, M., Anagnostopoulos, D.: Modeling collaborative processes with CMMN: success or failure? An experience report. In: Gulden, J., Reinhartz-Berger, I., Schmidt, R., Guerreiro, S., Guédria, W., Bera, P. (eds.) *Enterprise, Business-Process and Information Systems Modeling*, pp. 199–210. Springer, Berlin (2018)
  33. Rozman, T.: Teaching the software process modeling. In: IIS 2004 (2004)
  34. Russell, N., Ter Hofstede, A.H., Edmond, D., van der Aalst, W.M.: Workflow resource patterns. Technical report, BETA Working Paper Series, WP 127, Eindhoven University of Technology, Eindhoven (2004)
  35. Russell, N., Ter Hofstede, A.H., Edmond, D., van der Aalst, W.M.: Workflow data patterns: identification, representation and tool support. In: International Conference on Conceptual Modeling, pp. 353–368. Springer (2005)
  36. Russell, N., van der Aalst, W., ter Hofstede, A.: Workflow exception patterns. In: International Conference on Advanced Information Systems Engineering, pp. 288–302. Springer (2006)
  37. Schönig, S., Zeising, M., Jablonski, S.: Supporting collaborative work by learning process models and patterns from cases. In: 9th IEEE International Conference on Collaborative Computing: Networking, Applications and Worksharing, pp. 60–69. IEEE (2013)
  38. Stuit, M., Wortmann, H.: A collaboration process study with application of agent interaction and behavior diagrams. In: 2010 International Symposium on Collaborative Technologies and Systems, pp. 302–313 (2010)
  39. ter Hofstede, A.H., van der Aalst, W.M., Adams, M., Russell, N.: *Modern Business Process Automation: YAWL and Its Support Environment*. Springer, Berlin (2009)
  40. Villarreal, P.D., Lazarte, I., Roa, J., Chiotti, O.: A modeling approach for collaborative business processes based on the upcolbip language. In: International conference on business process management, pp. 318–329. Springer (2009)
  41. Wiemuth, M., Junger, D., Leitritz, M., Neumann, J., Neumuth, T., Burgert, O.: Application fields for the new object management group (OMG) standards case management model and notation (CMMN) and decision management notation (DMN) in the perioperative field. *Int. J. Comput. Assist. Radiol. Surg.* **12**(8), 1439–1449 (2017)
  42. Yan, Z., Reijers, H.A., Dijkman, R.M.: An evaluation of BPMN modeling tools. In: International Workshop on Business Process Modeling Notation, pp. 121–128. Springer (2010)
  43. Zensen, A., Küster, J.: A comparison of flexible BPMN and CMMN in practice: a case study on component release processes. In: 2018 IEEE 22nd international enterprise distributed object computing conference (EDOC), pp. 105–114. IEEE (2018)

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



**Ioannis Routis** is a Ph.D. candidate in the area of executable models for human-centric processes. He is a graduate of the Department of Informatics and Telematics of the Harokopio University of Athens, with a postgraduate diploma in advanced information systems in business administration. His Ph.D. focuses on human-centric processes modelling, as well as the requirements, in order to depict executable models. He has been a scientific associate in various European and national research projects. His research interests include human-centric processes management, social and collaborative processes modeling and the human view of Cyber-physical Human Systems in IoT. He has published papers in scientific journals, scientific books and international scientific conferences, and is also a reviewer in scientific journals.



**Mara Nikolaidou** is a Professor in the Department of Informatics and Telematics at Harokopio University of Athens. Her research interests include SoS engineering, system modeling and simulation, service-oriented architectures and BPM. Over the last years she actively participated in numerous research projects funded by national, European and international agencies on system engineering, service-oriented architectures, digital libraries and e-government. She is currently involved in

research projects focusing on SoS engineering, the Internet of Things, Cyber-physical Systems and Smart Cities. She has published more than 200 papers in international journals and conferences and actively participates in the organization of international conferences in the area of software and systems engineering. She is a member of IEEE SMC society and Systems Council.



**Dimosthenis Anagnostopoulos** received the B.Sc. and Ph.D. degrees from the Department of Informatics, University of Athens, Athens, Greece. He is a Professor of Information Systems and Simulation with the Department of Informatics and Telematics, Harokopio University of Athens, Kallithea, Greece, where he also served as the Rector. Since 2016, he has been the Dean of the Faculty of Digital Technology, Harokopio University of Athens. He is a Visiting Professor with Sussex

University, Brighton, U.K. His current research interests include information systems, E-government, semantic Web and Web services, modeling and simulation, and business process modeling. Mr. Anagnostopoulos is an Associate Editor of Requirements Engineering (Springer). He was elected as the Chair of the Greek Rectors Conference in 2014.