IET Software

Research Article

Modelling emergency response process using case management model and notation

ISSN 1751-8806
Received on 20th August 2016
Revised 14th January 2017
Accepted on 15th March 2017
E-First on 17th October 2017
doi: 10.1049/iet-sen.2016.0209
www.ietdl.org

Abobakr Y. Shahrah[1] ✉, *Majed A. Al-Mashari*[2]

[1]*Department of Software Engineering, King Saud University, Riyadh, Kingdom of Saudi Arabia*
[2]*Department of Information Systems, King Saud University, Riyadh, Kingdom of Saudi Arabia*
✉ *E-mail: shahrah@ksu.edu.sa*

Abstract: Emergency response is a knowledge-intensive process that is very hard to model and automate. This primarily returns to the unpredictability and unrepeatability nature that characterises such process. Traditional modelling approaches are too rigid and do not effectively support the flexibility and dynamicity required by emergency response process. Case Management Model and Notation (CMMN) is a new standard modelling specification that has recently been released by Object Management Group (OMG) to standardise the modelling of Case Management approach. The objective of this study is to demonstrate how the CMMN can be used to model emergency response process. The authors use the CMMN to build a template model for a generic emergency response process. This model can easily be extended or interchanged among different modelling tools or execution platforms. In addition, they present a case study of a flood management process as a concrete example of using CMMN in modelling emergency response process. Finally, they conclude that the CMMN has a great potential to model case-based processes such as emergency response, but CMMN still in its infancy. As a result, there is a lack of mature modelling tools and execution engines to execute the CMMN models.

1 Introduction

Emergency response is the most critical stage in Emergency Management. It is time-critical, knowledge-intensive, unstructured, very dynamic, and very difficult to pre-define all possible response scenarios [1–4]. In emergency response, exceptions are norms and each emergency incident presents unusual elements [5, 6]. Therefore, emergency response has to be scalable, flexible, and adaptable [7–9]. However, modelling and designing a software system to support the emergency response requirements are challenging [10, 11]. Furthermore, there is a research call in the domain of Business Process Management (BPM) for further domain-specific studies on how to apply process style approaches and techniques on various use cases applications (e.g. emergency response) [11, 12].

Emergency response becomes an interested application area for the Case Management approach [10, 13–15]. However, modelling emergency response process is very complicated, and this returns back to the unpredictable and unrepeatable nature of such process. Each emergency incident is unique and emergent especially in large-scale emergency events. Furthermore, each emergency incident requires different response activities and protocols based on the type, size, and complexity of the situation. Therefore, having a complete predefined model for all types of emergency response is impractical. Such case needs a flexible modelling approach that can be maintained and modified gradually. In addition, the model has to be adaptable to handle the exceptions and guide the knowledge workers (e.g. emergency responders). Unfortunately, traditional process-oriented modelling approaches (e.g. Business Process Model and Notation (BPMN)) fail in supporting such requirements [16–19].

Case Management approach with its accompanying modelling standard (Case Management Model and Notation (CMMN)) provides a flexible way to deal with the complexity of emergency response process. CMMN can be used to define the initial emergency response model based on the existing and known best practices, protocols, regulations or emergency operations plans (EOPs). Then, the CMMN model can easily be refined to accommodate any emergent requirements or adopt new best practices learned from previous cases. CMMN can also be used to define a solution template for developing a case-based emergency response system. This system acts as a guidance to the emergency response personnel in handling emergency incidents. Nevertheless, CMMN still a new and immature standard, and there is uncertainty about its applicability in real-world scenarios [20].

This paper aims to show how CMMN can be used to model emergency response process. This paper describes a CMMN modelling of a generic emergency response process as well as a case study of flood management process. In addition, this paper outlines the execution semantics of such model according to the OMG's CMMN standard specification. However, the execution considerations of the CMMN model is not in the scope of this paper due to the absence of mature execution platforms (engines).

The remaining of this paper is organised as follows: Section 2 provides a summary review on the existing process modelling approaches including CMMN. Section 3 outlines the related works that use process modelling for emergency response process. Section 4 explains how to use the CMMN to model a generic emergency response process. Section 5 discusses a case study of a flood management process as a concrete example of emergency response scenario. Section 6 briefly describes the execution semantics of the CMMN model. Finally, Section 7 concludes this paper by a summary of key findings and potential future work.

2 Background

Process modelling that supports model-driven execution can be generally classified into two main approaches: *activity-centric* and *data-centric* [21, 22]. Fig. 1 illustrates a simple and non-comprehensive taxonomy of these approaches. *Activity-centric* (process-centric) is the traditional business process modelling approach that can be *imperative* or *declarative* [23]. *Imperative activity-centric* approach is intended to support control flow of predictable and repeatable business process (routine). This approach focuses on the activities and their performing sequence (control flow) through an explicit definition of the process model at the design-time. Some examples of process models that follow the *imperative activity-centric* approach include Petri nets [24], BPMN [25], Event-driven Process Chains [26], and UML Activity Diagrams [27]. *Declarative activity-centric* approach is proposed

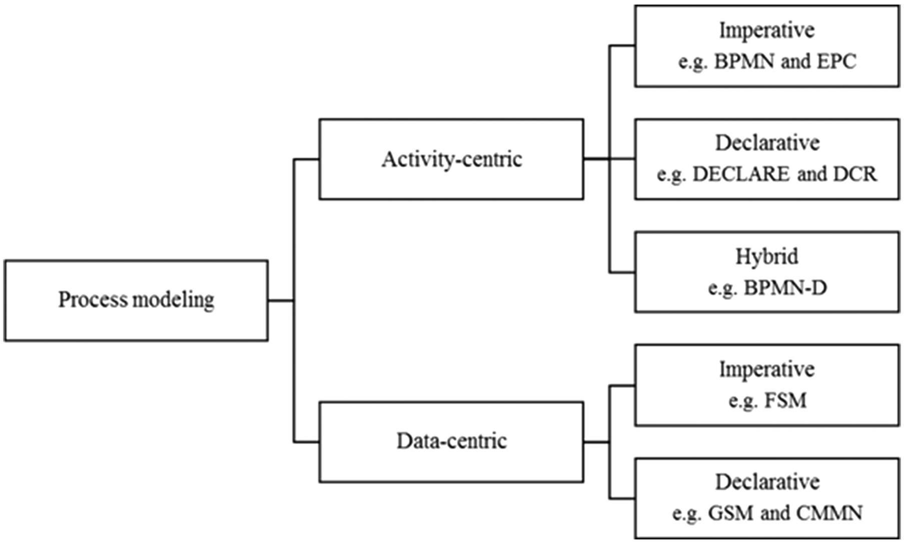

Fig. 1 *Process modelling approaches*

to enhance the flexibility of the traditional workflow management systems in supporting the loosely-structured processes such as DECLARE [28, 29] and DCR Graphs [30–32]. The researchers of [33, 34] proposed hybrid process modelling; i.e. combining some of *imperative* and *declarative activity-centric* approaches. For example, BPMN-D [35].

The second business process modelling approach is *data-centric* (information-centric). In this approach, the focus becomes more on the data structure and its lifecycle to drive the business process, and to deal with emergent events and requirements (event-driven). *Data-centric* approach is developed to enable more flexibility into the business process at the run-time in order to effectively handle unpredictable and unrepeatable cases. *Artefact-centric* (business artefacts) process model is a specific and popular type of *data-centric* approach that has an information (data) model and a lifecycle (behaviour) model [36, 37]. The lifecycle model can be *imperative* such as finite state machine (FSM), or *declarative* such as guard-stage-milestone (GSM) [38, 39]. GSM provided the basis for the development of CMMN specifications. Hinkelmann and Pierfranceschi [40] called for a hybrid *activity-centric* and *data-centric* modelling approach by combing both CMMN and BPMN. In contrast, Auer *et al.* [41] argued that a hybrid approach will not be suitable, and instead of that a full integration is needed.

CMMN is a specification that defines a common meta-model and notation for modelling a case in the context of the Case Management [42]. CMMN also defines an interchange format for exchanging case models among different tools. Object Management Group (OMG) released the CMMN 1.0 in May 2014 [43]. Later, in December 2016, OMG released the CMMN 1.1 that addresses the reported technical issues related to the first version (CMMN 1.0) [42]. The development of CMMN has been influenced by many ideas and concepts exist in the literature as well as some commercial products [44]. A short introduction and tutorial about CMMN 1.0 can be found in [45].

3 Related work

There are several case studies exist in the literature that use different process modelling approaches to model the emergency response. Rueppel and Wagenknecht [46] discussed the dynamic process modelling of emergency management activities. They proposed an approach that combines a meta-model of emergency management activities with a formal process model called ADEPT to support the dynamic changes of the process at the execution time. They demonstrated this approach on an operational flood scenario. Finally, they indicated that their future work will focus on the refinement of the meta-model as well as developing a prototype to evaluate their approach.

Fahland and Woith [47] presented an approach with declarative techniques to model adaptive processes that can change their behaviour at run-time for a disaster response. They used the concept of scenarios with a formal and operational semantics based on Petri nets and life-sequence charts to enable adaptation of process dynamics. However, they mentioned that the proof-of-concept run-time environment to implement their concepts and algorithms is one of the future works.

Kirsch-Pinheiro and Rychkova [48] explored the role of context data in Case Management, and proposed a context meta-model and architecture to enable a dynamic information representation. Then, they illustrated their findings on a flood crisis management process example using the FSM. They stated that this approach will help in integrating the contextual events and parameters into a process definition that can provide an automated guidance by suggesting the suitable activities for a particular situation.

Kushnareva *et al.* [49] proposed an intension-driven approach for modelling a crisis management process from goals to scenarios based on MAP and Statecharts formalisms. They stated that this approach helps in aligning the crisis management strategic level (goals) with the operational level (executable scenarios) through translating the MAP model into the Statecharts model. In other research works, Kushnareva *et al.* [50, 51] compared the activity-oriented (BPMN) and state-oriented paradigms (Statecharts) for supporting a crisis management process (flood scenarios). They concluded that the state-oriented modelling with Statecharts has a great potential for crisis management process, but the Statecharts notation needs an extension for the specifics of crisis management process.

As can be seen from the literature review described above, and to the best of our knowledge, we did not find a case study on the usage of CMMN in the emergency response scenario – only other related non-standardised approaches were used. Therefore, this work attempts to help in explaining the implementation of CMMN in the emergency response domain as well as, in general, evaluating the CMMN applicability and simplicity.

4 Emergency response process modelling

There are no universal standard activities for emergency response process. Emergency response process varies based on the type, size, and complexity of the emergency incident. Therefore, having a single and complete model for all types of emergency response process is infeasible. Similarly, responding to emergency incidents without generic guidance or initial plan is ineffective. Fortunately, CMMN provides the most effective and efficient solution to this issue. CMMN can be used to define a flexible model that contains the essential activities of emergency response process, and acts as a guiding template for emergency response personnel. This CMMN model can also be extended or used to develop a case-based emergency response system.

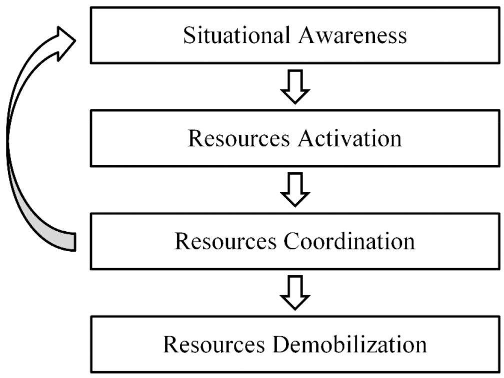

Fig. 2 *Basic emergency response stages*

In general, emergency response process follows four basic stages (shown in Fig. 2: *Situational Awareness, Resources Activation, Resources Coordination,* and *Resources Demobilisation*. Each stage has its own activities that vary based on the type, size, and complexity of the emergency incident. However, there are some common activities that are usually applicable for most of the emergency incident types. We use these activities to demonstrate how CMMN can be used to model the emergency response process. This model can also be extended for various incident types and more complicated emergency response scenarios.

The CMMN model for emergency response process is illustrated in Fig. 3. This model represents the basic emergency response stages (shown in Fig. 2) and possible associated activities by using the modelling notation of CMMN. The first `Stage` is the *Situational Awareness* which intended to identify, process, and comprehend the critical elements of information about an incident.

The CMMN decoration marks used in this `Stage` are `PlanningTable` (⊞), `Required` (!), and `Repetition` (#). The `PlanningTable` mark (⊞) refers to the scope of run-time planning of discretionary items (available inside the *Situational Awareness* stage) that can be considered for planning (e.g. *Fetch incident-related data* and *Send notification*). Discretionary items are optional items (stages or tasks) that can be defined in the planning phase (design-time), and these items can be activated by the case worker in the execution phase (run-time) according to his/her discretion. Discretionary items are depicted by a dashed line. The `Required` mark (!) specifies that the *Situational Awareness* `Stage` is required to complete or terminate before the containing `CasePlanModel` can complete (i.e. case completion). The `Repetition` mark (#) specifies that the *Situational Awareness* `Stage` will have repetitions (each repetition is a new instance of it).

The *Situational Awareness* `Stage` consists of three activities: *Gather incident information, Fetch incident-related data,* and *Send notification*. *Gather incident information* is a `Blocking HumanTask` to collect the basic information about the emergency incident. *Gather incident information* is automatically activated – i.e. there is no `ManualActivation` mark (>) – once the case instance is initiated. *Gather incident information* is an essential activity in the *Situational Awareness* `Stage`, therefore it is `Required` (!). As it is a continuing activity during the emergency response execution, we used the `Repetition` mark (#) to represent this behaviour. *Fetch incident-related data* is a discretionary `ProcessTask` to automatically extract any incident related data stored in internal or external databases like response guides, EOPs, standard operating procedures (SOPs), building blueprints and so on. This task can be repeated (#) each time the incident information is updated. *Send notification* is a discretionary `Non-blocking HumanTask` to send notification messages, like alerts or warnings, to the respected authorities or communities. This activity is manually activated (>) when required, and can be repeated multiple times (#). The completion of the *Situational Awareness* `Stage` (◊) triggers the achievement of the `Milestone` *Incident Recognised*. This `Milestone` is a prerequisite (◊) to the activation of the second emergency response `Stage` *Resources Activation*.

The *Resources Activation* `Stage` is used to activate available response resources and capabilities based on the identified incident requirements. This `Stage` is required (!), auto completed (▪), manually activated (>), and repeated (#). We use the `AutoComplete` (▪) in this `Stage` because it does not contain discretionary items and thereby it will be automatically completed after the completion of all its child elements (tasks). The *Resources Activation* `Stage` contains two basic activities: *Order resources* and *Mobilise resources*. *Order resources* is a `Blocking HumanTask` to request for response resources – like personnel, equipment, and tools – based on the type, size, and complexity of the emergency incident. This task is required (!), manually activated (>), and repeated (#). *Mobilise resources* is also a `Blocking HumanTask` to prepare, organise and deploy the requested response resources. It is also a required (!), manually activated (>), and repeated (#) task. The completion of the *Resources Activation* `Stage` triggers the achievement of *Resources Activated* `Milestone`.

The third emergency response `Stage` is the *Resources Coordination*. This `Stage` is used to plan response activities, and to manage response resources based on the type, size, and complexity of the emergency incident. This `Stage` has a `PlanningTable` (⊞) for the contained discretionary items. It is also manually activated (>) and repeated (#). The activities included in this `Stage` are: *Create Incident Action Plan (IAP), Update IAP*, and *Approve IAP*. IAP is an example of a standard incident management plan that can be used to formally document the incident goals, operational period objectives, response strategy and other details. IAP can be simple or very complicated based on the emergency incident characteristics. *Create IAP* is a discretionary and manually activated (>) `Blocking HumanTask` to develop the IAP. After the creation of the IAP, it has to be approved through the *Approve IAP*. *Approve IAP* is a repeated (#) `Blocking HumanTask` to authorise the publication and execution of the IAP. IAP is a dynamic document that can be updated or changed any time based on the emergent requirements, and this can be done by using the *Update IAP*. *Update IAP* is a manually activated (>) and repeated (#) `Blocking HumanTask`. Updates or changes to the initial IAP trigger the *Approve IAP*. The completion of the *Resources Coordination* `Stage` triggers the achievement of *Resources Coordinated* `Milestone`.

The fourth and last emergency response `Stage` is the *Resources Demobilisation*. This `Stage` includes the orderly, safe, and efficient return of response resources to its original location and status. It can only be manually activated (>) if the *Resources Activated* `Milestone` is achieved (◊). It is also required (!), auto completed (▪), and repeated (#). The *Resources Demobilisation* `Stage` contains one required (!), manually activated (>), and repeated (#) `Blocking HumanTask` to return the ordered and activated response resources. After the completion of the *Resources Demobilisation* `Stage`, the *Resources Returned* `Milestone` is triggered.

In addition to the basic emergency response stages, the incident in some cases might require additional activities like *Investigate the incident*. *Investigate the incident* is a discretionary `CaseTask` that can be used to call another case – i.e. opening a case for investigating the root causes or effects of the emergency incident. *Close incident* and *Cancel incident* are `UserEventListeners` ( ) that can be used by the end user to manually close or cancel the case at any point of time during the case lifecycle.

The CMMN model for emergency response process illustrated in Fig. 3 has an XML file representation according to the OMG's CMMN XML-Schema specification [42]. Fig. 4 shows an excerpt from the XML file representing the CMMN model for the emergency response process based on the CMMN 1.1 exchange format.

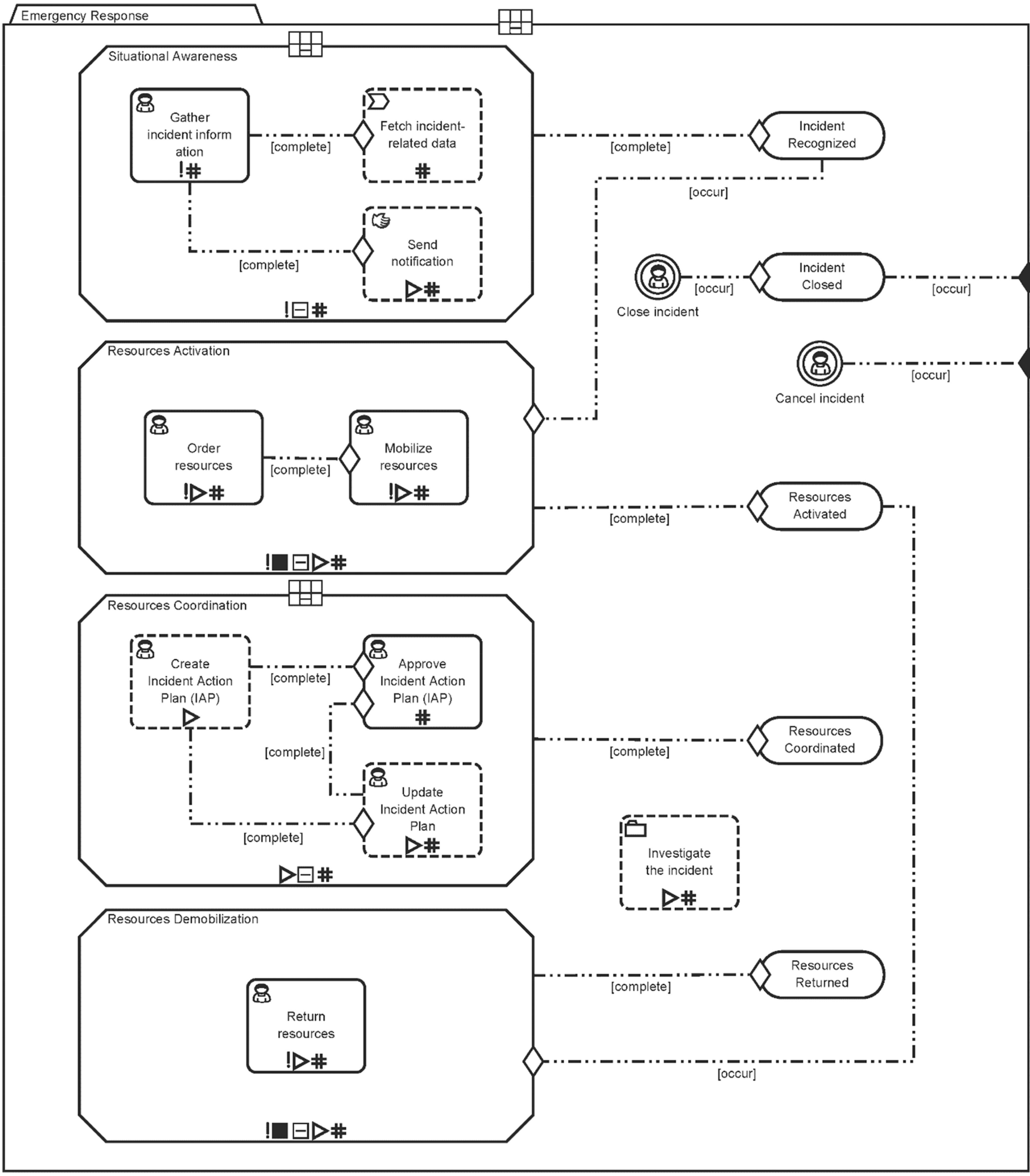

Fig. 3 *CMMN model for emergency response process*

5 Case study

In this section, we describe a case study of a concrete emergency response process example. This case study demonstrates how CMMN model can be used in more specific and realistic scenarios. The example is a flood management process based on the work of Kushnareva *et al.* in [50, 51]. This process is modelled by BPMN and implemented in the COS Operation Centre (COSOC) system. This system is developed by COS&HT company to manage the floods crises of Oka River in the Moscow region, Russia. Kushnareva *et al.* also presented an alternative model based on the Statecharts model. However, in this case study, we focus on the original BPMN model, and we show how it can be re-represented by CMMN instead of Statecharts model. Furthermore, we highlight some of the advantages of using CMMN over the Statecharts.

Floods on the Oka River are seasonal events, and threaten the living areas and critical infrastructure (e.g. railway bridge, a pontoon road bridge, an electric power plant, industrial storage facilities etc.). The emergency response personnel use the COSOC to collect the incident-related information, and to manage the response operations according to predefined emergency management guidelines. The flood management process is triggered when the water level of the Oka River reaches above 10 cm. The response actions are event-driven and escalated based on the development of the incident severity (e.g. reached water level, response units' reports etc.). Table 1 shows the different events and corresponding response activities supported by the COSOC flood

```
<cmmn:humanTask isBlocking="true" name="Gather incident information" id="PID__e9bff788-60b1-
40f1-b841-0795f55dca36">
                    <cmmn:defaultControl>
                        <cmmn:repetitionRule id="_636fbad4-69e9-4a71-a758-f6d5461fc3ad"/>
                        <cmmn:requiredRule id="_99e653f8-54ec-4b8e-990b-7c661e94289c"/>
                    </cmmn:defaultControl>
                </cmmn:humanTask>
                <cmmn:processTask isBlocking="true" name="Fetch incident-related data"
id="PID__62ac103c-9c64-4e94-af7f-0255dd9312ef">
                    <cmmn:defaultControl>
                        <cmmn:repetitionRule name="sdfg" id="a8701480-53ff-4ac1-a202-
4690c29a5876">
                            <cmmn:condition language="dfg" id="_0f94905e-b7f3-4ea0-a082-
c9e7f3449d06">fgdf</cmmn:condition>
                        </cmmn:repetitionRule>
                    </cmmn:defaultControl>
                </cmmn:processTask>
                <cmmn:humanTask isBlocking="false" name="Send notification" id="PID__7f3e1c81-
0590-469a-90a1-7aa85ff6b978">
                    <cmmn:defaultControl>
                        <cmmn:repetitionRule id="ae682503-4a76-446b-8d99-bdb62f911cef"/>
                        <cmmn:manualActivationRule id="_216fbe59-2daa-4ddb-a08e-
bffcdbfa9121"/>
                    </cmmn:defaultControl>
                </cmmn:humanTask>
            </cmmn:stage>
```

Fig. 4 *An excerpt from the XML file representing the CMMN model for the emergency response process*

Table 1 Events and response activities of COSOC BPMN-based flood management process [50, 51]

Stage	Event		Response	
	ID	Description	ID	Description
0	E0	flood alert: $h > 10$ cm	T1	set temporary barriers
	E1	end of alert	—	end of Alert
1.1	E2	emergency: $h > 10$ cm and keeps rising	T2	state of emergency
			T3	alert people
			T4	evacuate people
1.2	E3	elevated risk: $h > 25$ cm	T5	provide boats, pontoons
			T6	supply with water
	E4	request for resources	T7	request reinforcement
	E5	report: resources are sent	—	resources are sent
	E6	high risk: $h > 40$ cm	T12	limit traffic/create deviation
			T10	shut down electric power station
	E7	request for evacuation	T4	evacuate people
	E8	alert: $h > 45$ cm	T13	close bridge
	E9	request for rescue operation	T8	rescue operations
	E10	alert: streets are flooded	T9	pump out water from streets
	E11	alert: electric power plant is flooded	T11	secure electric power station
2	E12	below critical: $h < 25$ cm	T14	remove barriers
			T15	organise reconstructions
	E13	end of emergency	—	end of emergency

management process (actors are ignored because they cannot be modelled in CMMN).

Fig. 5 illustrates the CMMN model for the flood management process of this case study. The model includes three main stages: *S0. Flood Alert, S1. Flood Emergency,* and *S2. Restoring Normal Functioning*. These stages are initiated according to the triggering event that represents the incident escalation (i.e. increased water level). The first activity in the model is *Get water level*, which is depicted as a `ProcessTask`. The purpose of this task is to gather the real-time information of water level from the environment monitoring resources (e.g. social networks, wireless sensors, video cameras etc.). Based on the collected information, the situation is evaluated through a `DecisionTask` called *Evaluate the situation*. This task invokes a DMN model that contains the events and their corresponding decisions (response activity) based on the reported water level (E0, E2, E3, E6, E8, and E12). For example, the event *E0. Flood Alert: $h > 10$ cm* initiates the first `Stage` *S0. Flood Alert,* which includes one `Blocking HumanTask` called *T1. Set temporary barriers*. In case of the water level goes down again within 12 h, the emergency alert is ended E1 (end of emergency). This can be modelled in CMMN as a `TimerEventListener` ( ) to catch predefined elapses of time.

The second `Stage` *S1. Flood Emergency* is initiated when the event E2 (Emergency: $h > 10$ cm and keeps rising) occurs. This `Stage` has two sub-stages *S1.1. Preparation* and *S1.2. Emergency Control* to deal with different scenarios of flood management based on a variety of event escalations. For example, in case of E6 (high risk: $h > 40$ cm), two `Blocking HumanTasks` must take place: *T12. Limit traffic/Create deviation* and *T10. Shut down electric power station*. Another example is the *T11. Secure electric power station* which is a `Blocking HumanTask` that must be carried out when either the event E8 (alert: $h > 45$ cm) or E11 (alert: electric power plant is flooded) occurs. E4, E9, E10, and E11 are `UserEventListener` ( ) to catch events that are raised by a user. A new construct introduced in this case study is the `CaseFileItem` ( ) which represents a piece of information of any nature (structured/unstructured; or simple/complex). This element is used to model the report of E5 (resources are sent).

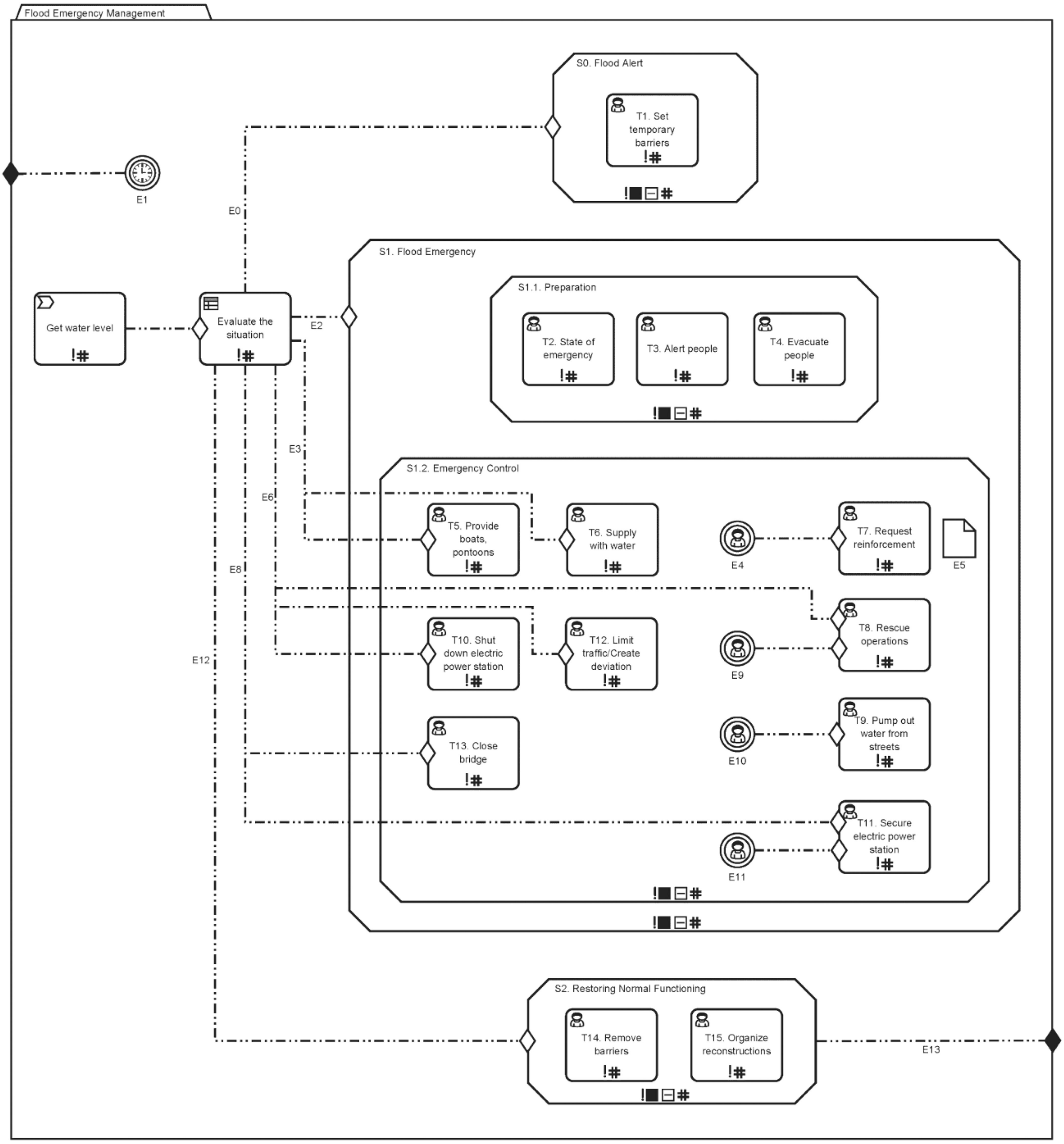

Fig. 5 *CMMN model for flood management process*

Finally, the third `Stage` *S2. Restoring Normal Functioning* is initiated when the water level goes under the critical level E12 (below critical: $h < 25$ cm). This `Stage` includes two `Blocking HumanTasks`: *T14. Remove barriers* and *T15. Organise reconstructions*. After completing this `Stage`, the incident emergency is ended E13 (end of emergency) and the case can be closed.

As a conclusion, we found that there are several advantages of CMMN over the Statecharts recommended by Kushnareva *et al.* in [50, 51]. First, CMMN is a standardised modelling approach which is developed and maintained by a professional organisation (OMG). This definitely encourages the industry vendors to implement and support, especially the ones who have contributed to the development of CMMN specifications. Second, CMMN supports the invocation of a decision table modelled by DMN through a `DecisionTask`. This helps in separating the decision logic (i.e. business rules) from the process logic (i.e. business activities) as recommended by process management best practices. Furthermore, this simplifies the process modelling and encourages a centralised business rules management. Lastly, Statecharts model focuses on the states modelling and does not explicitly show the tasks (activities), however, CMMN model does. Furthermore, CMMN has the feature of modelling discretionary items (`Stage` or `Task`) which helps in defining the scope of run-time planning for a certain context.

6 Execution semantics

The execution semantics of CMMN is described by the lifecycle of important CMMN element instances, including `Task`, `Stage`, `Milestone`, `EventListener`, and `CaseFileItem` [42]. In addition to that, there are behavioural property rules that also describe the behaviour of a case management engine. These rules include Applicability rule, Stage.autocomplete, ManualActivationRule, RequiredRule, and RepetitionRule.

A `Case` instance contains both information model (`caseFileModel`) and a behavioural model (`casePlanModel`). In addition, there are roles (`caseRoles`) that correspond to humans who could participate in handling the `Case`. The `caseFileModel`, `casePlanModel`, and `caseRoles` are all initialised when the `Case` instance is created, and the `Stage` instance that implements the `caseFileModel` starts executing in an Active state. While the `Case` instance is not in a closed state, the `caseFileModel` can be modified, planning can occur, and human participants can be assigned to roles. A detailed description about the CMMN elements' lifecycles, instance states, and transitions is available in the OMG's CMMN 1.1 specification standard document [42].

7 Conclusion and future work

The characteristics of emergency response represent a knowledge-intensive, unstructured, and dynamic process. Dealing with unpredictability and unrepeatability are some of those characteristics. This implies that modelling emergency response needs a flexible and adaptable approach to handle exceptions and accommodate emergent requirements. Traditional process-oriented modelling approaches (like BPMN) lack such capabilities, and therefore are not suitable to model Case Management work (like emergency response process). CMMN is a new modelling approach recently introduced by OMG to address such limitations, and support the concepts and principles of Case Management approach.

This paper briefly introduced the different process modelling approaches including CMMN, and demonstrated how CMMN can be used to model a generic emergency response process. This paper also showed a sample of XML file representation for the CMMN model that can be used for model interchange among different modelling tools or execution platforms. Furthermore, this paper presented and discussed more specific case study based on a concrete example of flood management process. The execution of CMMN model has been briefly discussed to understand the run-time behaviour of CMMN elements.

Despite CMMN is a promising modelling approach for Case Management, it is still in its early adoption stages by industry vendors. This probably returns to the technical issues associated with the first version of the CMMN standard (CMMN 1.0). Recently, OMG has done great efforts to resolve the reported issues, and released a new version of CMMN standard (CMMN 1.1), which will hopefully be more refined and stable. Consequently, there is still a lack of mature modelling tools and execution platforms to support the implementation of CMMN; but we believe that the near future is promising. Our future work will focus on the refinement of the presented model as well as the execution considerations on a mature run-time platform.

8 References

[1] Kushnareva, E., Rychkova, I., Le Grand, B.: 'Modeling business processes for automated crisis management support: lessons learned'. IEEE Ninth Int. Conf. on Research Challenges in Information Science, 2015

[2] Mendonça, D.: 'Decision support for improvisation in response to extreme events: learning from the response to the 2001 World Trade Center attack', *Decis. Support Syst.*, 2007, **43**, (3), pp. 952–967

[3] Di Ciccio, C., Marrella, A., Russo, A.: 'Knowledge-intensive processes: characteristics, requirements and analysis of contemporary approaches', *J. Data Semant.*, 2014, **4**, (1), pp. 29–57

[4] Jul, S.: 'Who's really on first? A domain-level user, task and context analysis for response technology'. Proc. of the 4th Int. Conf. on Information Systems for Crisis Response and Management, ISCRAM 2007, 2007

[5] Turoff, M., Chumer, M., de Walle, B.V., *et al.*: 'The design of a dynamic emergency response management information system (DERMIS)', *J. Inf. Technol. Theory and Appl. (JITTA)*, 2004, **5**, (4), p. 3

[6] Lindell, M.K.: 'Emergency management', in Bobrowsky, P.T. (Ed.): '*Encyclopedia of natural hazards*' (Springer Netherlands, 2013)

[7] Turoff, M.: 'Past and future emergency response information systems', *Commun. ACM*, 2002, **45**, (4), pp. 29–32

[8] U.S. Department of Homeland Security: '*National response framework*' (U.S. Department of Homeland Security, 2013, 2nd edn.)

[9] Turoff, M., White, C., Plotnick, L.: 'Dynamic emergency response management for large scale decision making in extreme hazardous events', in Burstein, F., Brézillon, P., Zaslavsky, A. (Eds.): '*Supporting real time decision-making*' (Springer USA, 2011)

[10] Hill, J.B., Dunie, R., Chin, K.: '*Critical capabilities for BPM-platform-based case management frameworks*' (Gartner, Inc., 2015)

[11] Hofmann, M., Betke, H., Sackmann, S.: 'Process-oriented disaster response management: a structured literature review', *Bus. Process Manag. J.*, 2015, **21**, (5), pp. 966–987

[12] Hofmann, M., Sackmann, S., Betke, H.: 'Using workflow management systems to improve disaster response processes'. 27th Int. Conf. on Advanced Information Networking and Applications Workshops (WAINA), 2013, 2013

[13] Swenson, K.D.: '*State of the art in case management*' (Fujitsu America, Inc., 2013, White paper edn.)

[14] Ward-Dutton, N.: '*The case for smarter case management*' (MWD Advisors, 2013)

[15] Le Clair, C., Moore, C.: '*Dynamic case management – an old idea catches new fire*' (Forrester Research, 2009)

[16] Zhu, W.-D., Benoit, B., Jackson, B., *et al.*: '*Advanced case management with IBM case manager*' (IBM Redbooks, 2014, 4th edn.)

[17] Van der Aalst, W.M.P., Weske, M., Grünbauer, D.: 'Case handling: a new paradigm for business process support', *Data Knowl. Eng.*, 2005, **53**, (2), pp. 129–162

[18] Kaan, K., Reijers, H., van der Molen, P.: 'Introducing case management: opening workflow management's black box', in Dustdar, S., Fiadeiro, J., Sheth, A. (Eds.): '*Business process management*' (Springer Berlin Heidelberg, 2006)

[19] Reichert, M., Weber, B.: '*Enabling flexibility in process-aware information systems: challenges, methods, technologies*' (Springer Science & Business Media, 2012)

[20] Kurz, M., Schmidt, W., Fleischmann, A., *et al.*: 'Leveraging CMMN for ACM: examining the applicability of a new OMG standard for adaptive case management'. Proc. of the 7th Int. Conf. on Subject-Oriented Business Process Management, 2015

[21] Marrella, A., Mecella, M., Russo, A., *et al.*: 'A survey on handling data in business process models (discussion paper)'. 23rd Italian Symp. on Advanced Database Systems (SEBD), 2015

[22] de Man, H.: '*Case management: A review of modeling approaches*' (BPTrends, 2009)

[23] Fahland, D., Lübke, D., Mendling, J., *et al.*: 'Declarative versus imperative process modeling languages: the issue of understandability'. Enterprise, Business-Process and Information Systems Modeling: 10th Int. Workshop, BPMDS 2009, and 14th Int. Conf., EMMSAD 2009, held at CAiSE 2009, Amsterdam, The Netherlands, 8–9 June 2009

[24] van der Aalst, W.M.P.: 'The application of petri nets to workflow management', *J. Circuits Syst. Comput.*, 1998, **08**, (01), pp. 21–66

[25] 'Business Process Model and Notation (BPMN), Version 2.0.2', 2013

[26] Mendling, J.: 'Event-driven process chains (EPC)', '*Metrics for Process models: empirical foundations of verification, error prediction, and guidelines for correctness*' (Springer Berlin Heidelberg, 2008)

[27] 'OMG Unified Modeling Language (OMG UML), Version 2.5', 2015

[28] Pesic, M., Schonenberg, H., Aalst, W.M.P.v.d.: 'DECLARE: full support for loosely-structured processes'. 11th IEEE Int. Enterprise Distributed Object Computing Conf., 2007, EDOC 2007, 2007

[29] van der Aalst, W.M.P., Pesic, M., Schonenberg, H.: 'Declarative workflows: Balancing between flexibility and support', *Comput. Sci. – Res. Dev.*, 2009, **23**, (2), pp. 99–113

[30] Debois, S., Hildebrandt, T., Slaats, T.: 'Hierarchical declarative modelling with refinement and sub-processes'. 12th Int. Conf. Business Process Management, BPM 2014, Haifa, Israel, 7–11 September 2014

[31] Hildebrandt, T., Marquard, M., Mukkamala, R., *et al.*: 'Dynamic condition response graphs for trustworthy adaptive case management'. On the Move to Meaningful Internet Systems: OTM 2013 Workshops, 2013

[32] Slaats, T., Mukkamala, R., Hildebrandt, T., *et al.*: 'Exformatics declarative case management workflows as DCR graphs', in Daniel, F., Wang, J., Weber, B. (Eds.): '*Business process management*' (Springer Berlin Heidelberg, 2013)

[33] De Smedt, J., De Weerdt, J., Vanthienen, J.: 'Multi-paradigm process mining: retrieving better models by combining rules and sequences'. On the Move to Meaningful Internet Systems: OTM 2014 Conf.: Confederated Int. Conf.: CoopIS, and ODBASE 2014, Amantea, Italy, 27–31 October 2014

[34] Westergaard, M., Slaats, T.: 'Mixing paradigms for more comprehensible models'. 11th Int. Conf. Business Process Management, BPM 2013, Beijing, China, 26–30 August 2013

[35] De Giacomo, G., Dumas, M., Maggi, F.M., *et al.*: 'Declarative process modeling in BPMN'. 27th Int. Conf. Advanced Information Systems Engineering, CAiSE 2015, Stockholm, Sweden, 8–12 June 2015

[36] Nigam, A., Caswell, N.S.: 'Business artifacts: an approach to operational specification', *IBM Syst. J.*, 2003, **42**, (3), pp. 428–445

[37] Hull, R.: 'Artifact-centric business process models: brief survey of research results and challenges'. On the Move to Meaningful Internet Systems: OTM 2008: OTM 2008 Confederated Int. Conf., CoopIS, DOA, GADA, IS, and ODBASE 2008, Monterrey, Mexico, 9–14 November 2008

[38] Hull, R., Damaggio, E., Fournier, F., *et al.*: 'Introducing the guard-stage-milestone approach for specifying business entity lifecycles'. Web Services and Formal Methods: 7th Int. Workshop, WS-FM 2010, Hoboken, NJ, USA, 16–17 September 2010

[39] Hull, R., Damaggio, E., Masellis, R.D., *et al.*: 'Business artifacts with guard-stage-milestone lifecycles: managing artifact interactions with conditions and events'. Proc. of the 5th ACM Int. Conf. on Distributed Event-based System, 2011

[40] Hinkelmann, K., Pierfranceschi, A.: 'Combining process modelling and case modelling'. 8th Int. Conf. on Methodologies, Technologies and Tools enabling e-Government, MeTTeG14, 2014

[41] Auer, D., Hinterholzer, S., Kubovy, J., *et al.*: 'Business process management for knowledge work: considerations on current needs, basic concepts and models', in Piazolo, F., Felderer, M. (Eds.): '*Novel methods and technologies for enterprise information systems*' (Springer International Publishing, 2014)

[42] 'Case Management Model and Notation (CMMN), Version 1.1', 2016

[43] 'Case Management Model and Notation, Version 1.0', 2014

[44] Marin, M., Hull, R., Vaculín, R.: 'Data centric bpm and the emerging case management standard: a short survey'. Business Process Management Workshops, 2013

[45] Marin, M.A.: 'Introduction to the case management model and notation (CMMN)', ArXiv e-prints, 2016

[46] Rueppel, U., Wagenknecht, A.: 'Improving emergency management by formal dynamic process-modelling'. 24th Conf. on Information Technology in Construction, 2007

[47] Fahland, D., Woith, H.: 'Towards process models for disaster response'. Business Process Management Workshops: BPM 2008 Int. Workshops, Milano, Italy, 1–4 September 2008

[48] Kirsch-Pinheiro, M., Rychkova, I.: 'Dynamic context modeling for agile case management'. On the Move to Meaningful Internet Systems: OTM 2013 Workshops, 2013

[49] Kushnareva, E., Rychkova, I., Deneckére, R., *et al.*: 'Modeling crisis management process from goals to scenarios'. AdaptiveCM 2015 – 4th Int. Workshop on Adaptive Case Management and Other Non-Workflow Approaches to BPM, 2015

[50] Kushnareva, E., Rychkova, I., Grand, B.L.: 'Modeling business processes for automated crisis management support: lessons learned'. 2015 IEEE 9th Int. Conf. on Research Challenges in Information Science (RCIS), 2015

[51] Kushnareva, E., Rychkova, I., Le Grand, B.: 'Modeling and animation of crisis management process with statecharts'. 14th Int. Conf. Perspectives in Business Informatics Research: BIR 2015, Tartu, Estonia, 26–28 August 2015