



HAL
open science

A Collaborative Information System Architecture for Process-Based Crisis Management

Omar Tahir, Chihab Hanachi, Christophe Sibertin-Blanc, Éric Andonoff,
Frederick Benaben, Vincent Chapurlat, Thomas Lambolais

► **To cite this version:**

Omar Tahir, Chihab Hanachi, Christophe Sibertin-Blanc, Éric Andonoff, Frederick Benaben, et al.. A Collaborative Information System Architecture for Process-Based Crisis Management. 12th International Conference on Knowledge-Based Intelligent Information and Engineering Systems (KES 2008), Sep 2008, Zagreb, Croatia. pp.630-641, 10.1007/978-3-540-85567-5_78 . hal-00353788

HAL Id: hal-00353788

<https://hal.science/hal-00353788>

Submitted on 7 Jun 2023

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A Collaborative Information System Architecture for Process-Based Crisis Management

Omar Tahir¹, Eric Andonoff¹, Chihab Hanachi¹, Christophe Sibertin-Blanc¹,
Frédéric Benaben², Vincent Chapurlat³, and Thomas Lambolais³

¹ University of Toulouse, IRIT Laboratory, 2 rue du Doyen Gabriel Marty,
31042 Toulouse Cedex, France

{tahir, andonoff, hanachi, sibertin}@univ-tlse1.fr

² DRGI, Ecole des Mines d'Albi-Carmaux, Route de Teillet, 81013 ALBI Cedex 9, France
{Frederick.Benaben}@enstimac.fr

³ Laboratoire de Génie Informatique et d'Ingénierie de Production (LGI2P)
Parc Scientifique G. Besse, site EERIE-EMA, 30035 Nîmes cedex 1, France
Vincent.Chapurlat@ema.fr, Thomas.Lambolais@ema.fr

Abstract. In computer-based crisis management, participating actors have to put in common their information systems (IS). One of the main issues they have to face is to interoperate and coordinate these ISs in order to ease their collaborations and take the right decisions at the right moment. For that, they have to set up a Collaborative Information System (CIS) for orchestrating a Collaborative Process (CP) whose activities are performed by the participating ISs. The aim of this paper is to define the functional requirements of such a CIS and then to propose a conceptual architecture meeting these requirements. This architecture supports a perception-decision-action iteration for crisis reduction. Indeed, it provides means to capture information about the impacted real world, to support the definition and the adaptation of the CP managing the crisis reduction, and to coordinate and assign to each participant the actions to be undertaken. These actions modify the real world and lead to perform a new iteration until the crisis resolution.

Keywords: Architecture, Collaborative Information System, Collaborative Process, Adaptiveness, Crisis Management.

1 Introduction

In crisis situations, participating actors/organizations have to act simultaneously and in emergency for reducing the crisis and its impacts on the real world. To achieve this common goal efficiently, these actors must collaborate or at least act in a coordinated way in order to make their activities as efficient as possible.

Crisis coordination is a difficult task since it requires to take into account activities' distribution and actors' autonomy (they decide by themselves how to perform their missions and services) and to combine their competences in a flexible way in order to meet the evolving requirements of the crisis. In order to support such complex coordination, organizations have developed several computer-based tools such as GIS [9] or Collaborative Tools [5].

For instance, in the framework of the IsyCri ANR¹ (French Research Agency) project, we collaborate with the “Préfecture” of the Tarn² in order to develop a Collaborative Information System helping them to manage crisis. Within this framework, as soon as a crisis occurs, a crisis cell is created and led by the ”Préfet”³. This crisis cell is made up of the representatives of the different actors implied in the crisis resolution [6]. In such a context, the coordination role of the crisis cell is to elaborate a collaborative process, for crisis response, synthesizing different plans of actions proposed by the participating partners (police, military forces, medical organizations...). Doing so, the crisis cell avoids activities’ inconsistency and redundancy, and ensures the convergence of the different activities towards the common goal. This need for coordination is even important in cases of crisis that occur in an ill-structured context. For instance, in humanitarian crisis, the actors’ coordination helps to maximize the services they offer to the injured persons, refugees, and so on, but also to reduce time of their interventions and to increase their reactivity. Besides, Non Governmental Organizations (NGO) or organizations in charge of coordination such as OCHA (Office for the Coordination of Humanitarian Affairs), systematically have been using platforms for information exchange and sharing (e.g. <http://www.reliefweb.int>). Then, the participating actors can better position their missions both geographically but also with regard to the activities they undertake

This work is developed in the framework of the IsyCri ANR project [6]. It adopts a computer-based approach to support coordination and it is based on three assumptions: (i) a Collaborative Information System (CIS) should be set up to support crisis resolution, (ii) the partners’ Information Systems should be integrated to the CIS, and (iii) crisis resolution is process-oriented: a Collaborative Process (CP) is defined for specifying the coordination of partners’ activities.

The coordination of the actions of the different partners involved in a crisis resolution is supported by the CIS which can be seen as an Inter-Organizational Information System [8] acting as a middleware component between the ISs of the involved partners.

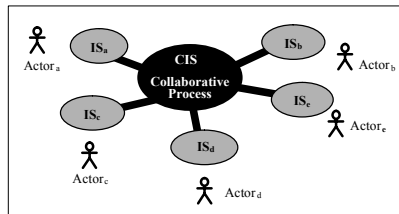


Fig. 1. Collaborative Information System

The CIS (see figure 1) is responsible for the management (definition, adaptation and orchestration) of the CP and it requires the partners’ ISs to perform the missions or services they have in order to reduce the crisis. Doing so, the different partners master their missions and services and consequently the autonomy of their ISs is preserved.

¹ Agence Nationale de la Recherche.

² French Department.

³ In France, a “Préfet” is a high-ranking civil servant who represents the State at the level of a department or a region. The “Préfet” is at the head of the “Préfecture”.

The aim of the paper is to propose a CIS architecture for crisis management. More precisely, the main research questions addressed in this paper are: which are the requirements that a CIS must meet? And how does a CIS architecture for crisis management look like?

The proposed architecture supports a perception-decision-action iteration for crisis reduction. Indeed, it provides means to capture information about the impacted real world, to support the definition and the adaptation of the collaborative process managing the crisis reduction, and to coordinate and assign to each participant the actions to be undertaken. These actions modify the real world and lead to a new iteration, where these tasks are performed again, taking into account the new crisis context.

The paper is organized as follows. Section 2 presents the requirements of a collaborative process-based information system. In section 3, we give the CIS conceptual architecture. First, we provide an overview of how the partners' ISs can be integrated with the CIS. Then, we detail its internal architecture by giving its functional components, its kinds of users (roles) and its databases. Section 4 presents the dynamics of our architecture, i.e. how its different components interact, through several UML sequence diagrams. Section 5 discusses our proposition and concludes the paper.

2 Requirements for a Collaborative Information System

The IsyCri project aims at developing a framework and a software prototype namely the CIS for crisis management. From the study of historical information about crisis, the way they have been managed, and on the basis of meetings, interviews and workshops with actors involved in crisis cells (the potential CIS' users), we have identified four major objectives to be met by the CIS:

- Providing an overview on the crisis and its evolution.
- Providing reliable information about the means (plans, processes, resources ...) that are available to resolve the crisis.
- Ensuring communication interoperability: increasing interaction capability between the information systems of the implied partners.
- Allowing the coordination of the activities of the actors that intervene in the crisis resolution: supporting the orchestration of the dynamic Collaborative Process followed for managing the crisis.

Hereafter, we present the main functional requirements of the CIS. However, for the real system (and not for the prototype), non functional requirements are of paramount importance: in particular, reliability, safety and real-time constraints purposes should be taken into account, since the system is designed to run in a very disrupted environment (with unreliable data connections in particular). In particular, the CIS supports the Collaborative Process, helps organization partners in the crisis management, but it cannot make any decision on their behalf. Actors remain the pilots, in any circumstances.

Let us precise some vocabulary and the kinds of users involved in a CIS in order to clarify the functional requirements. The *Impacted Real World* is the observed system. The *Treatment System* is the added-on system set up to resolve the crisis. It includes Actors and their ISs, the CIS, and the Collaborative Process. A *Crisis Model* is a

representation of a crisis in accordance to a Crisis meta-model given in [6]. A *Crisis Cell* is a group of people in charge of the crisis management by using the CIS. Three kinds of users (roles) of a CIS have been identified. A crisis cell generally includes one person who is globally and ultimately responsible for the proper management of the crisis (e.g. in our project, the “Préfet”); his activity corresponds to the *Pilot* role, that can be fulfilled by any person delegated for this. A crisis cell also includes one person for each of the civil institutions involved in the crisis management (e. g. the police, firemen, water supply, hospital...); this person is the representative of his institution and stands as the *Partner* role. Finally, we also consider the *Crisis Model Manager* role that handles the observed information about the situation coming from the partners’ organizations and is responsible for updating the Crisis Model according to these information.

From the previous objectives assigned to the CIS, we have derived four main functional requirements to be met. The CIS should give means to:

- *Manipulate and share a dynamic crisis representation.* Four services should be provided for that purpose. First, the CIS should give access to the crisis model and views on it to Partners. Second, it should help the partners, via simulation or analysis, to assess the situation of the Impacted Real World. Third, the CIS should give means to update the Crisis Model. Finally, the CIS should offer tools for validating and verifying the crisis model (notably its consistency).
- *Define, adapt, configure and enact the crisis collaborative process.* All these functions are to be performed by the Pilot of the CIS. The definition consists in creating the collaborative process from the Crisis meta-model which contains both descriptions of past crisis and the way they have been resolved. This Collaborative process is obviously adapted according to the evolution of the crisis. The configuration consists in granting rights to the different partners in order to allow them to perform some activities of the Collaborative Process. Finally, the enactment corresponds to the execution of the Collaborative Process, where the CIS should have means to access to the partners’ Information Systems in order to extract information or submit activities to be performed. Except for the adaptation function, the other ones could be supported by a conventional Workflow Management System [1].
- *Manage data.* This function is offered to each Partner. It should give means to retrieve data about the impacted real world, to build scoreboards in order to provide synthetic information for risks evaluations, and to simulate the consequences of possible actions.
- *Administrate the CIS.* This function allows the Pilot to define, configure and modify the parameters required to connect the partners’ Information Systems to the CIS and give them means to communicate with one another (chat, forum...).

3 Conceptual Architecture of a Collaborative Information System

The purpose of this section is to present the conceptual architecture of a CIS for crisis management. This architecture is independent from any implementation consideration. A CIS connects the different ISs of partners involved in a crisis resolution. This architecture is presented in top-down way. First, we give an overview of how the partners’ ISs can be integrated with the CIS. Then, we give a more detailed view of

the CIS focusing on its internal architecture and presenting its functional components, and the involved databases. The dynamic aspects will be presented latter.

3.1 Integration of the Partners' ISs with a CIS

Figure 2 shows the three components involved during a crisis: the Impacted Real World, a set of ISs belonging to the different partners involved in the crisis resolution and a CIS which coordinates the actions undertaken by these partners.

More precisely, the CIS manages a Collaborative Process (CP) which describes the orchestration of the actions to be undertaken in order to reduce the crisis. Moreover, the CIS does not directly access or act on the impacted real world but rather uses the partners' ISs to do that. In fact, the partners' IS are the "eyes and hands" of the CIS on the impacted real world. Indeed, in a crisis situation, the impacted real world is initially observed. This observation, which is done via the partners' ISs, is modelled and represented in the CIS by the *Crisis Model*. From this *Crisis Model* and using what we call the *Treatment Model*, which models a set of actions (and their coordination) to be undertaken when a given situation or a given event is detected, a CP is then built in response to the crisis. This CP is then orchestrated by the CIS which orders the different participating partners to perform actions. These actions modify the real world which is observed at any time. Then, the CP can possibly be adapted or totally changed according to the new perception of the crisis, as mentioned, it is an iterative process.

Figure 2 also shows the three activities of a CIS: *Observation* of the Impacted Real World (Observe and get Observations arrows), *Deliberation* in order to build a CP which is convenient to reduce the crisis (Deliberate arrow) and *Act* on the Impacted Real World (Ask for Performing Actions and Perform Actions arrows). These three activities are repeated until the resolution of the crisis.

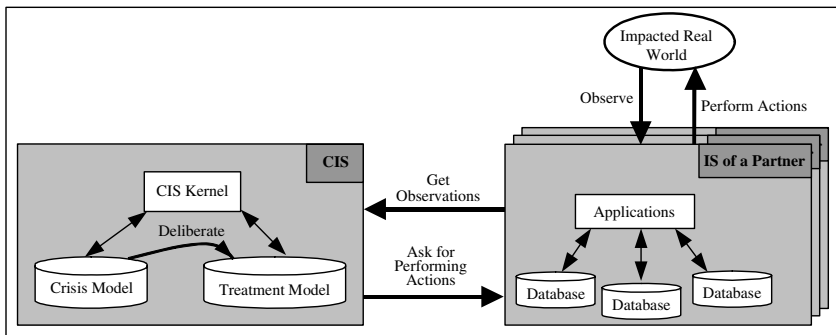


Fig. 2. Global View of a CIS

The CIS kernel is composed of a set of functional components which manage the CP, the Crisis Model and the Treatment System and support the Deliberation activity. These components will be detailed in the next section.

3.2 Internal View of a CIS

We now focus on the internal architecture of a CIS presenting the involved databases, the functional components and the interfaces that make it up. Figure 3 below presents this internal architecture, which meets the functional requirements presented in section 2 and is organized around:

- 3 databases, each one represented by a cylinder.
- 7 functional components, each one represented by a rectangle.
- 10 interfaces supporting the interaction between functional components and kind of users. Interfaces are represented by arrows.

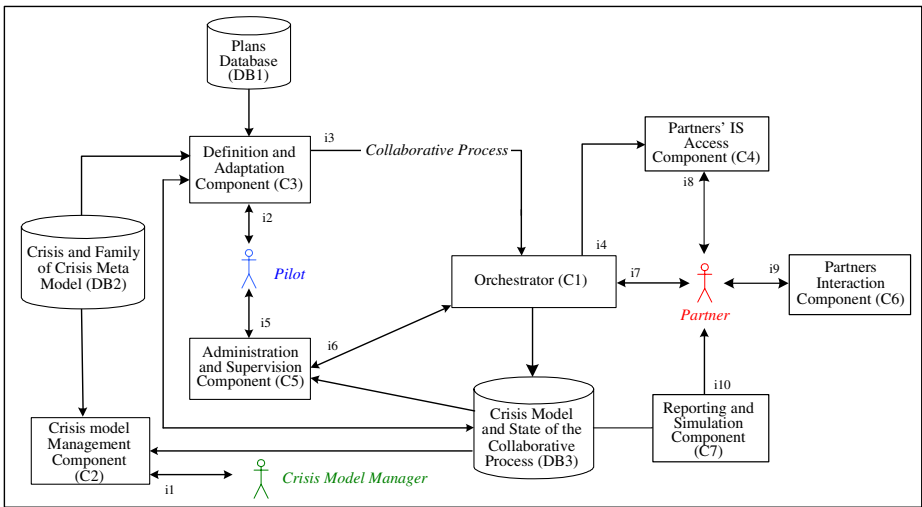


Fig. 3. Internal View of a CIS

We distinguish three databases, DB1, DB2, DB3, storing respectively:

- A set of processes, each one defining a set of coordinated activities to be undertaken when a given situation or event occurs (Treatment Model of figure 2).
- A Crisis and Crisis Family Meta Model.
- The actual representation of the impacted real world (Crisis Model of figure 2) and the current state of the CP.

DB3 is used to have an accurate representation of both impacted real world and undertaken actions, while DB2 and DB1 are used for the Deliberation activity in order to build a convenient CP for crisis reduction. In fact, DB2 and DB1 are built from crisis ontologies [6] defining respectively the crisis concepts, their links and a crisis classification (civil, humanitarian, social...) regarding DB2, and, regarding DB1, the plans and actions to undertake in order to resolve each type of crisis. Concerning DB3, the crisis model corresponds to an instance of these ontologies. These latter have been implemented in OWL (<http://www.w3.org/TR/owl-features/>) using Protégé software (<http://protege.stanford.edu/>).

Regarding the functional components, the *Orchestrator* component (C1) is the main CIS component. Its role is to orchestrate the CP. It consults the partners' ISs via interface I4, interacts with a partner to ask for the execution of services via interfaces I7 and I8, and receives reports on the execution state of the requested services via interface I7. The Orchestrator remains under the control of the Pilot through the *Administration and Supervision* component (C5), at the same time as it stores the state of the CP in the DB3 database. The *Crisis Model Management* component (C2) serves to build and maintain a model of the crisis, as reliable and up-to-date as possible in the DB3 database. The *Definition and Adaptation* component (C3) permits the Pilot to build the CP at the beginning of the crisis, but also to check this CP and adapt it according to the evolution of the Crisis Model (DB3) and the Treatment Model (DB2 and DB1).

The *Administration and Supervision* component (C5) aims at giving rights to partners, monitors the orchestration of the CP and makes the guidance of CP execution by the pilot possible (alternative, choice, ...). The *Partners' IS Access* component (C4) allows the CIS to be connected with the partners' ISs. It transmits to the partners the requests for missions and services execution. The *Partners Interaction* component (C6) provides communications services (chat, forum...) helping to take the appropriate decision on the processing system including the CP. Finally, the *Reporting and Simulation* component (C7) can be used to visualize both the impacted real world and the CP state. It can also be used to simulate the crisis evolution and its consequences.

4 The Dynamics of the CIS: How to Use the CIS to Manage the Crisis

Figure 3 shows the functional architecture of a CIS: the main services it offers to the crisis cell members and the information it needs to manage. In this section we present its dynamics, that is how the components of this architecture are used and interact to manage a crisis. First of all, we have to notice that the controlled management of a crisis includes two sub-processes:

- A *Collaborative Process* (CP) performed by the *operational* agents of the Partners' organizations, which act on the field to secure population and goods, prevent the occurrence of new disastrous events, repair the damages, rescue people in danger and so on.
- A *Control Process* performed by the members of the crisis cell, that consists in collecting information about the actual situation of the crisis and giving appropriate orders to the operational agents of the partners' organizations in charge of fighting the crisis concretely; in fact, the control process is a kind of meta-process since it aims at defining the Collaborative Process (CP), adapting or reshaping it according to the evolution of the situation and supervising its execution.

The CIS does not directly intervene in the crisis field. Instead, it is the tool used by the crisis cell members to manage the crisis control process. Thus, the main part of the SIC's dynamics is related to the way it supports the crisis control process.

A description of this control process is shown in Figure 4 as a UML2 interaction overview diagram [12]. It is an abstract view of the process that stands at the same level as the functional architecture in Figure 3. Once the crisis cell for the crisis

management is set up, the control process includes five tasks: two for managing the crisis model (a representation of the impacted real world) shared by the crisis cell members, two for defining and adapting the structure of the CP according to the crisis model, and one for supervising the CP.

The *Crisis_Model_Initialization* task collects information about the crisis from the partners and, among the known crisis models stored in the DB2 database (Crisis and Family of Crisis meta-model), selects the one that best matches to the situation. The *Crisis_Model_Management* task aims at maintaining the crisis model in conformance with information about the impacted real world; in case of an expected evolution, it is just a matter of updating the values of some parameters or variables, while in case of an evolution that changes the nature of the crisis, a new crisis model has to be built in the same way than in the initialization task. In both cases, the Orchestration task presented below is stopped.

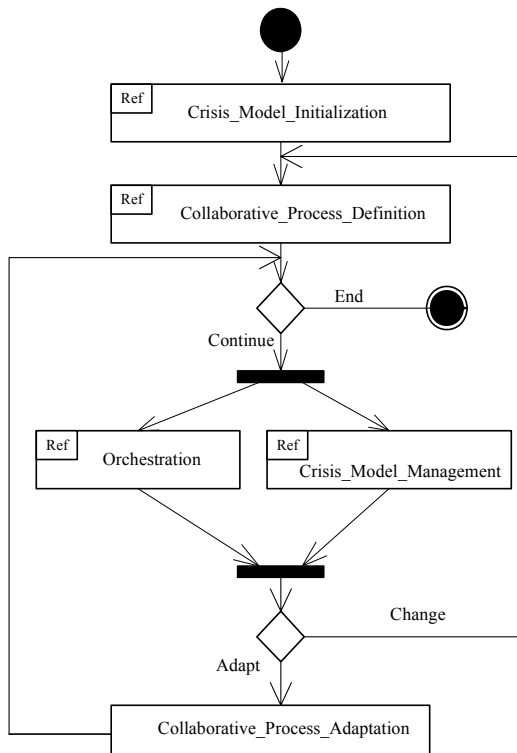


Fig. 4. UML overview diagram of the Control Process

The *Collaborative_Process_Definition* task consists in finding the most appropriate actions to reduce the crisis; the crisis model gives the expected results of these actions, while the DB1 database (Plan database) contains the action plans and means that can be enacted by the partners according to their respective responsibilities. Thus, defining the CP is a matter of finding the best match between what should be done to

reduce the crisis and what can be done by the partners, and to coordinate the activities the partners are requested to perform.

The *Collaborative_Process_Adaptation* task accounts for the updates of the crisis models that require only little changes in the structure of the CP such as removing or adding actions, using information to select the most relevant alternative or reordering some actions in a more positive way.

Finally, the *Orchestration* task is in charge of supervising the execution of the CP. It is performed by the Orchestrator component, a workflow engine that sends to the partners (or partners' Information Systems) the actions they have to perform and maintains the current state of the CP thanks to the execution reports supplied by the partners.

The structure of the control process described in Figure 4 should be read as follows. Once the *Crisis_Model_Initialization* is achieved by the Crisis Model Manager, the Pilot defines the initial CP. Then, the process *Continues* and, concurrently, the Orchestrator orchestrates the execution of this CP while the Crisis Model Manager deals with the available information to update the Crisis Model. In the *Adapt* case of a simple update, the orchestration of the CP is stopped in order to be updated and then a new < orchestration // Crisis_Model_Management > loop is started. In case of a *Change* in the Crisis Model, the orchestration of the CP is also stopped, and the Pilot enters a new performance of the *Collaborative_Process_Definition* task. If the resulting new CP is empty then the *End* signal provokes the termination of the control process since the crisis is considered as being solved, otherwise the *Continue* signal makes the control process enter in the loop again.

We have specified UML Sequence Diagrams that define standard scenarios of how the members of the crisis cell and the components of the SIC interact to perform each of these tasks.

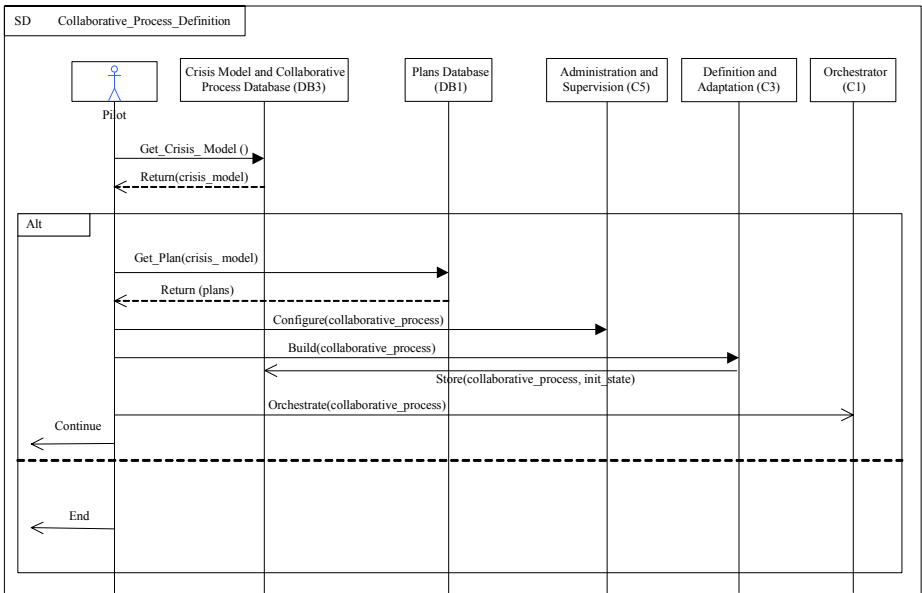


Fig. 5. Sequence Diagram of the Collaborative_Process_Definition task

Because of space limitation, we only present here two of these sequence diagrams for the Collaborative_Process_Definition and the Crisis_Model_Management tasks. These sequence diagrams are respectively visualized in figures 5 and 6. They are self-explanatory and we only comment on the included frames that appear in them. In figure 5, the *Alt* indication in the upper left corner of the included frame introduces an alternative between the two parts separated by the horizontal dashed line; that means that, according to the result returned by the DB3 database, either the Pilot builds and stores an effective collaborative process and emits a *Continue* signal, or he just emits an *End* signal that produces the termination of the control process. The same holds in the sequence diagram of figure 6, where the task ends either with a *Change* or with an *Adapt* signal.

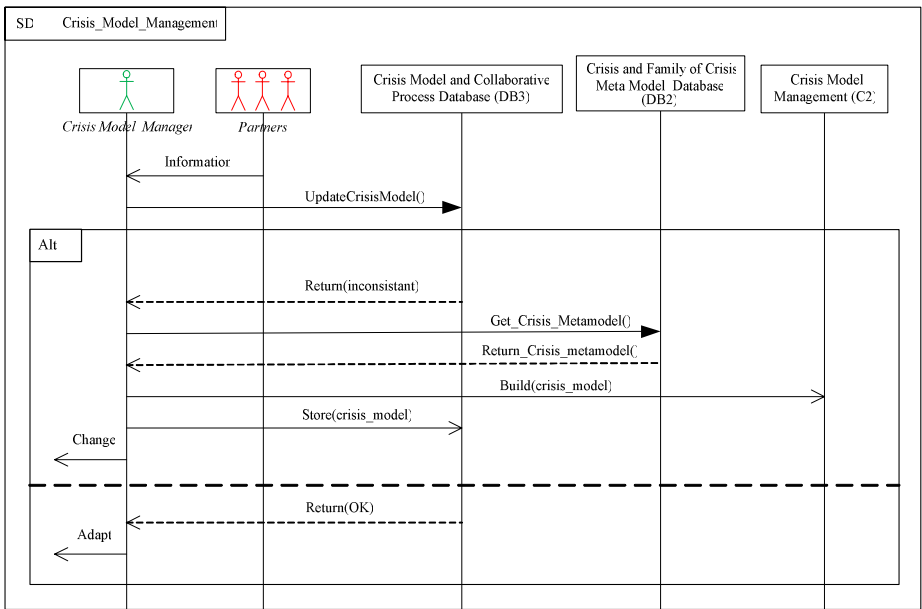


Fig. 6. Sequence Diagram of the Crisis_Model_Management task

5 Discussion and Conclusion

This paper has described a Collaborative Information System architecture for process-based crisis management. This architecture features three main advantages. First, it supports the collaborative-process design, enactment and adaptation. *Adaptation* is an innovative aspect regarding traditional workflow management systems [1] and very few works address this aspect [2]. In our architecture, adaptation is supported at two different levels: the crisis model may be changed and the collaborative process can be transformed. The second advantage is that *Context-awareness* is made possible thanks to databases able to record and structure information about the crisis, its evolution and possible reactions plans. Third, interfaces are defined to connect partners' information

systems to the CIS. Regarding this aspect, interoperability issues in the context of Services Oriented Architecture are discussed in [6].

Regarding the state of the art, in the Crisis Information System field, even if a lot of information systems have been developed to support crisis management, most of them are specific to a type of crisis [3], focus on the influence of innovative tools [4], or are limited to simulation or geographic information management and visualization [7]. Collaborative tools (such as CSCW) have also been developed in the Crisis field [5]. Most often they are used specially to support interactions among partners and document sharing among them. Nevertheless, such tools do not adopt a process perspective. In this way, the closest work to ours is [11], developed in the context of the WORKPAD project (<http://www.workpad-project.eu>). In this work, a P2P architecture is designed from users requirements, and it includes data storage and communication, middleware and user layers. This architecture also manages processes and workflow patterns mining is made possible. What makes our architecture different from the one proposed in [11] is that our collaborative process is a first class citizen and also our architecture remains conceptual and does not impose any technological choice (P2P, Web Services, Grid, etc). Although our architecture is independent from any implementation consideration, a SOA approach fits in it [6]. The collaborative process may invoke activities implemented as web services registered in an UDDI and coordinated by an orchestration engine. An implementation on top of the open source PETALS [14] Platform is in progress. PETALS is an infrastructure that implements a SOA architecture with the Enterprise Service Bus (ESB) technology. On the one hand, PETALS supports mediation and interoperability between heterogeneous software components (thanks to adapters), and, on the other hand, it provides service orchestration engine based on standards like Business Process Execution Language (BPEL). In our context, PETALS supports the interaction between the partners' information systems, and also orchestrates the collaborative process. At design time the crisis collaborative process is defined with BPMN (Business Process Modeling Notation), which is a high level description language, while at run time it is transformed onto BPEL [13] which is an operational language and moreover supported by PETALS.

Although our conceptual framework is a solid basis for an implementation, there are some open issues, notably crisis process adaptation. This issue assumes a very deep study of the parameters to be adapted in a process. It may concern the process control structure, the organizational model inherent to the process, allocations policies of tasks to actors, among other aspects. The complexity of this issue is due to the fact that all these adaptations may be investigated in a dynamic context, i.e. at run time during the process execution.

Acknowledgments

This work has been supported by the ANR (French Research Agency) through the IsyCri project. The authors would like to thank the other project partners namely Thales Communication and EBM Websourcing.

References

1. van der Aalst, W., van Hee, K.: *Workflow Management, Models, Methods, and Systems*. MIT Press, Cambridge (2004)
2. Adams, M., ter Hofstede, A., Edmond, D., van der Aalst, W.: A Service-Oriented Implementation of Dynamic Flexibility in Workflows. In: 14th Int. Conf. on Cooperative Information System, Montpellier, France, pp. 291–308 (2006)
3. de Addams-Moring, R.: Tsunami Self-evacuation of a Group of Western Travelers and Resulting Requirements for Multi-Hazard Early Warning. In: 4th Int. Conf. on Information System for Crisis Response and Management, Delft, The Netherlands (2007)
4. Avery-Gomez, E., Turoff, M.: Interoperable Communication: an Analysis of SMS Text-Message Exchange. In: 4th Int. Conf. on Information System for Crisis Response and Management, Delft, The Netherlands (2007)
5. Cai, G.: Extending Distributed GIS to Support Geo-Collaborative Crisis Management. *Geographic Information Science* 11(1), 4–14 (2005)
6. Couget, P., Benaben, F., Lauras, M., Hanachi, C., Chapurlat, V.: Information Systems Interoperability as a way to Partners Integration in a Crisis Context. In: 20th Int. Conf. on Software and Systems Engineering and their Applications, Paris, France (2007)
7. Dilekli, N., Rashed, T.: Towards a GIS Data Model for Improving the Emergency Response in the Least Developing Countries: Challenges and Opportunities. In: 4th Int. Conf. on Information System for Crisis Response and Management, Delft, The Netherlands (2007)
8. Divitini, M., Hanachi, C., Sibertin-Blanc, C.: Inter Organizational Workflows for Enterprise Coordination. In: Omicini, A., Zambonelli, F. (eds.) *Coordination of Internet Agents*, pp. 46–77. Springer, Heidelberg (2001)
9. Erl, T.: *Service-Oriented Architecture: Concepts, Technology and Design*. Prentice-Hall, Englewood Cliffs (2005)
10. Kumar, V., Bugacov, A., Coutinho, M., Neches, R.: Integrating Geographic Information Systems, Spatial Digital Libraries and Information Spaces for conducting Humanitarian Assistance and Disaster relief Operations in Urban Environments. In: 7th International Symposium on Advances on Geographic Information Systems, Kansas City, Missouri, USA (1999)
11. de Leoni, M., de Rosa, F., Marrella, A., Mecella, M., Poggi, A., Krek, A., Manti, F.: Emergency Management: from User Requirements to a Flexible P2P Architecture. In: 4th Int. Conf. on Information System for Crisis Response and Management, Delft, The Netherlands (2007)
12. Object Management Group UML 2.0 Specification (2005), <http://www.omg.org>
13. Ouyang, C., van der Aalst, W., Dumas, M., ter Hofstede, A.: From BPMN Process Models to BPEL Web Services. In: 4th Int. Conf. on Web Services, Chicago, USA, pp. 285–292 (2006)
14. PETALS (2008), <http://petals.objectweb.org/>