Process-aware Information Systems for Emergency Management

Massimiliano de Leoni[1], Andrea Marrella[2], and Alessandro Russo[2]

[1] Technical University of Eindhoven, Eindhoven, The Netherlands
m.d.leoni@tue.nl
[2] SAPIENZA - Università di Roma, Rome, Italy
{marrella,arusso}@dis.uniroma1.it

Abstract. Nowadays, Process-aware Information Systems (PAISs) are widely used in many business scenarios, e.g., by government agencies, by insurance companies, and by banks. Despite this widespread usage, the typical application of such systems is predominantly in the context of business scenarios. Nevertheless, emergency management can also benefit from the use of PAISs; for instance, the metaphor of a business process fits very good with the concept of emergency recovery plan. This paper summarizes an invited talk given by the first author for the EMSOA'10 workshop that has been co-located with the ServiceWave 2010 Conference. This paper starts the basic PAIS' requirements for the domain of emergency management, then it gives an overview of the nowadays' literature on using PAISs for Emergency Management. Finally, the paper proposes an architecture and a system to support the execution of emergency management processes.

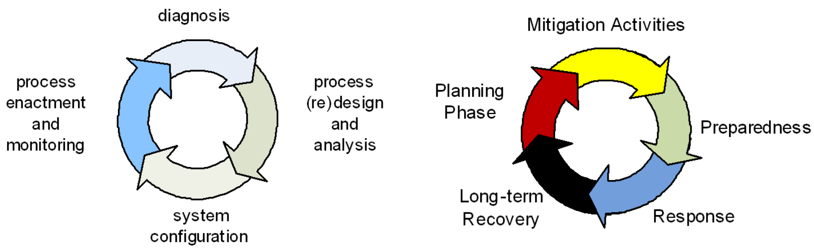
1 Introduction

Nowadays organisations are always trying to improve the performance of the processes they are part of. It does not matter whether such organisations are dealing with classical static business domains, such as loans, bank accounts or insurances, or with pervasive and highly dynamic scenarios. The demands are always the same: seeking more efficiency for their processes to reduce the time and the cost for their execution.

According to the definition given by the Workflow Management Coalition[1], a workflow is "the computerised facilitation of automation of a business process, in whole or part". The Workflow Management Coalition defines a Workflow Management System as "a system that completely defines, manages and executes workflows through the execution of software whose order of execution is driven by a computer representation of the workflow logic".

So a PAIS is driven by some process model. The model may be implicit or hidden, but the system supports the handling of cases in some (semi-)structured form. PAISs also have in common that they offer work to resources (typically people). The elementary pieces of work are called *work items*, e.g., "Approve travel request XYZ1234". These work items are offered to the users by the so-called *work-list handler*. This system component takes care of work distribution and authorization issues. Typically, PAISs use a so-called "pull mechanism", i.e., work is offered to all resources that qualify and

[1] Workflow Management Coalition Web Site - http://wfmc.org



(a) Business Process Management (b) Emergency Management

Fig. 1. The comparison of the respective life cycles

the first resource to select the work item will be the only one responsible for its execution.

On the basis of several studies of emergency plans [7, 9] and some end-users interviews we have conducted [8], it becomes evident that emergency plans are, in fact, similar to business processes. Therefore, a correct introduction of a PAIS during emergency management can result in improving the efficiency and effectiveness in dealing with the emergency's aftermath, thus reducing the event's consequence. As a further confirmation of the equivalence of emergency plans and usual processes, we can compare the life cycles of a typical PAIS [14] and of the emergency management [12]. The phases of a PAIS life-cycle can be roughly mapped onto emergency management as follows:

1. *Process Designing*, i.e. when the process schemas are prepared by experts, coincides with the *Preparedness Phase*, where contingency plans are create to manage the aftermath of emergencies
2. *System Configuration* can be mapped partially upon the *Preparedness* for the definition of the skills the process requires, and partially upon the *Response Phase*, for what concerns the configuration of the running instances as regards to the actual emergencies. The *Response Phase* during Emergency Management concerns the period when prepared plans are enacted to manage actual emergency occurrences.
3. *Process Enactment and Monitoring* corresponds particularly to the *Response Phase*, but also the Long-term Recovery. The *Long-term Recovery Phase* is the period after *Response Phase*, when all urgencies have been managed and, subsequently, it is aimed to restore the living condition to a situation comparable to that before the emergency breaks out. However, it is worthy highlighting that the most of the *Long-term Recovery Phase* is usually executed with no support of PAISs. Indeed, this phase involves so many external entities and services that a central PAIS would be strongly limiting.
4. The *diagnosis phase* is very similar to the planning phase and the *Mitigation Activities*. Indeed, during the Diagnosis Phase, the execution logs are analyzed in order evaluate the management of the past emergency, thus finding bottlenecks and, hence, proposing process improvements.

In sum, the process metaphor is suitable for dealing with emergencies and, as stated in [15] "*past and future objective remain the same in crises: providing relevant communities collaborative knowledge systems to exchange information*".

2 User Requirements

In work [9], Jul reflects the American disaster management practices, investigating how the emergency size influences the response type and how collaboration should occur on the spot to deal with the aftermath of an emergency. The American disaster management requirements reported in this work are also confirmed by the experience of the WORKPAD project [2] and a joint analysis with a German civil-protection officer [7].

Jul starts classify calamities in three groups: *(i)* simple local emergencies that are short-lived event whose effects are localized in a single community; *(ii)* disasters, i.e. long-lived event affecting many communities, but community and response infrastructures are affected by few damages; *(iii)* catastrophes, long-lived event affacting hundreds of communities, destroying almost every infrastructure and damaging the response systems. For example, the three category comprise respectively an house explosion, 9/11 terroristic attack and 2005 Hurricane Katrina. From this moment on, we are not going to consider any longer local emergencies, since they do not require an extensive PAIS support: the necessity of some collaboration is quite limited, due to the local nature of the happening.

As far as the Response and Communication Infrastructure, this can be characterized by local damages (medium-size disasters) or extensive destruction (large catastrophes). Even if it is not disrupted, past experiences suggest the existing infrastructure should be used as less as possible. For instance, the Katrina catastrophe has shown that if all civil-protection units use the existing infrastructure, it is destined to collapse or to experience a too low performance level, due to the overload. Indeed, it was not designed to support so many users at the same time. Therefore, **it is advisable to opt for Mobile-ad hoc Networks** (Requirement 1), which are wireless networks in which hosts act both as end points sending/receiving packets and as relays forwarding packets along the correct nodes' paths towards the intended recipients [1].

The support for context awareness is crucial: this is confirmed by [9]: "*context can be characterized by their similarity to environment known to the user, with individual contexts being very familiar, somewhat familiar or unfamiliar*". Moreover "*a given user, particularly in larger events, is likely to work in a variety of contexts, either because of physical relocation or because of changes in the context itself*". Therefore, users cannot be assumed to have local knowledge of the geography and resources of the area. Consequently, **PAISs should be integrated with Geographic Information Systems (GISs), which allow users to gain a deep knowledge of the area, or, more in general, with Geographic-aware Information Systems** (Requirement 2).

As from [9], "*context may be more or less austere*" and "*operations may be established in novel locations*", as well as "*response activities may be relocated*". Hence, "*uncertainty and ambiguity are inherent to disaster*", from which it follows that response technology must allow for flexibility and deviation in their application, while imposing standard structures and procedures. So, **PAISs should allow for large process specifications that are specialized time by time according to the specific happenings** (Requirement 3), as well as **they need to foresee techniques to adapt the process execution to possibly changing circumstances and contingencies** (Requirement 4).

Last but not least, PAIS Client Tools must be extremely usable and intuitive. As from [9], '*'the response typically involve semi-trained or untrained responders ... and the proportion of semi-trained and untrained responders increases with the scale of the event, and they assume greater responsibility for response activities*.". As Emergency Management Systems are not used on daily basis but in exceptional cases, even experts could be not very trained: training sessions could be helpful, but a real emergency is totally different. From this, follows that the **Emergency Management Systems should be so intuitive that they can be easily mastered after few interaction sessions** (Requirement 5).

3 Survey of the current-day PAIS' approaches for Emergency Management

Current approaches based on the adoption of PAISs in the emergency management domain mainly aim at providing support for the *preparedness*, *response* and *recovery* activities. In order to support emergency operators in quickly and efficiently defining process models, in [10] the authors propose a domain-driven process adaptation approach based on configurable process models. Configurable process models capture and combine common practices and process variants related to specific emergency domains, which can be configured in a specific setting leading to individualized process models. In a configurable model, different process variants are integrated and represented through *variation points*, enabling process configuration. Variation points allow removing part of a large process specification that are irrelevant for the current enactment, thus meeting Requirement 3 of Section 2. During process configuration, the requirements stemming from a specific emergency scenario reduce the configuration space, but due to the number of variation points and constraints, the model is in general too difficult to be manually configured. Process configuration is thus performed using interactive questionnaires which allow process experts to decide each variation point by answering a question

A different approach based on design-time synthesis from scenarios is proposed in [5]. Under this approach, small processes, named *scenarios* and modeled as Petri Nets, are dynamically merged upon request. Thus, a large emergency management process is synthesized by composing several fragments. Specifically, in each process representing a *scenario*, places are associated with labels, which determine the points where scenarios can be concatenated: places with the same label can be merged. This is a different approach for Requirement 3, but it is also useful for meeting Requirement 4: if needed, new scenarios can even be appended at run-time as a mechanism of adaptation of system and process behavior.

The support of process adaptation needs for emergency management is crucial (as for Requirement 4); a similar requirement already exists for classical business process management. Therefore, there is a large body of work on this topic. The only existing approaches that are applicable for emergency management are the *automatic* approaches, since *manual* ones, where an expert adapts manually the process upon contingencies, would delay the execution in a way that can lead to consequences (e.g., death of people, collapsing of buildings).

Inside the category of the automatic approaches, the *pre-planned* strategies foresee that, after designing the process schema, the designers describe the policies for the management of all possible discrepancies that may occur. As a consequence, the number of possible discrepancies needs to be known a priori, as well as the way to deal with their occurrence. Therefore, *pre-planned* adaptation is feasible and valuable in static contexts, where exceptions occur rarely, but it is not applicable in dynamic scenarios where policies for too many discrepancies should be designed. On the other side, *unplanned* automatic adaptation approaches try to devise a general recovery method that should be able to handle any kind of event, including those unexpectable. The process is defined as if exogenous events (i.e., contingencies) cannot occur. Whenever discrepancies are detected thus leading to no successful process termination, the control moves to a single exception handler in charge of activating a general recovery method. Such a method is intended to modify the failing executing process into a new one that achieves all goal of the old but such that it terminates in the changed environment.

Nowadays, commercial and open source PAISs use a *manual* adaptation approach (e.g., ADEPT2, ADOME, AgentWork), a *pre-planned* approach (e.g., YAWL), or both (e.g., COSA, Tibco, WebSphere, OPERA), and there exists no PAIS using an *unplanned* adaptation approach (for a more comprehensive analysis see also [4]). An interesting previous case study for using PAISs for emergency management has been carried on using the AristaFlow BPM Suite [11]. AristaFlow, the commercial version of the ADEPT2 framework, allows for verification of the process structure and it features an intuitive approach to adapt process instances at run-time to deal with contingencies. This enables non-computer experts to apply changes and adapted processes are checked for soundness. But, unfortunately, the approach is still manual, even though interesting work has been conducted to simplify the work of adapting process instances. AristaFlow aims also at meeting Requirement 2 of Section 2: relevant information linked to tasks is visualized on geographic maps of the area where the emergency has broken out.

AristaFlow provides also a mobile version for the Work-list Handler [13] that can be installed on Smartphones based on Windows Mobile. The idea is that rescue operators connect to the server to retrieve the tasks they are assigned to and, later, they execute such tasks while disconnected from the PAIS server. Finally or at any point in time, end users can synchronize their work with the server. The problem of this approach is that tasks are executed off-line and, hence, previous tasks can modify at any time the list of tasks that need to be executed afterwards. Consequently, users can be assigned to and carrying on tasks that, when synchronizing after finishing executing them, are learnt to be no more required. The most appropriate solution is that the server itself is constantly being available on the spot and running on mobile devices, so that off-line solutions are prevented. Section 4 describes ROME4EU, a mobile Process-aware Information System that allows for on-line task assignment and execution.

An automatic process adaptation approach based on execution monitoring has been proposed in [4]. Adaptation through execution monitoring requires to define: *(i)* techniques for monitor and possibly predict discrepancies between the internal *virtual* reality built by the PAIS and the external *physical* reality built "sensing" the real world; *(ii)* techniques for identification of corrective actions to deal with the new execution context; *(iii)* techniques for automatic process restructuring in order to successfully terminate the adapted process in the new context. According to the proposed approach, the PAIS assigns tasks to resources considering execution context and resources' capabilities. For each execution step, an *execution monitor* aligns the internal *virtual* reality built by the PAIS with the *physical* reality and data retrieved from external world by sensors (intended as any software and/or hardware component able to get contextual information), possibly adapting the process to unforeseen exogenous events and producing an adapted process to be executed.

4 The WORKPAD Approach

In complex emergency/disaster scenarios teams from various emergency-response organizations collaborate with each other to achieve a common goal. In these scenarios the use of smart mobile devices and applications can improve the collaboration dynamically. The WORKPAD project [8], finished in 2009, aimed at the development of an Adaptive Peer-to-Peer Software Infrastructure for supporting the collaborative work of human operators in Emergency Management Scenarios.

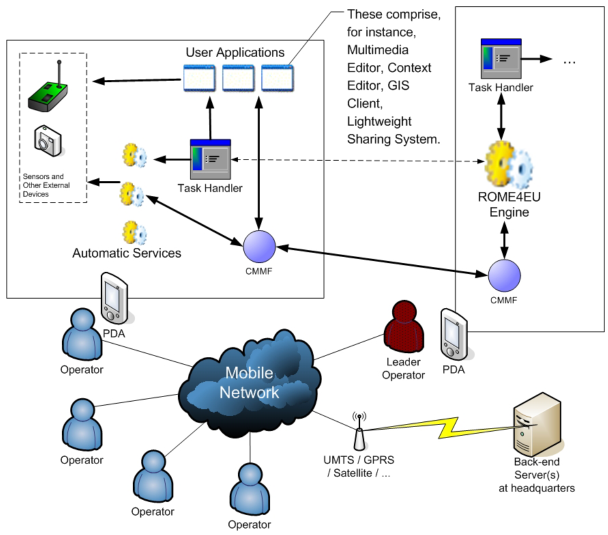


Fig. 2. The High-Level Infrastructure of the WORKPAD System

According to the initial user requirements collected [8], WORKPAD has developed a two-level infrastructure: a first level is deployed on the spot and includes the diverse rescue teams that are sent to the area in order to be actively included in providing assistance to the involved people and in mitigate the aftermath. There are several front-end teams on the field, each composed by rescue operators and headed by a "leader operator", who coordinates the intervention of the other team members. Rescue operators are equipped with PDAs and their work is orchestrated by a WfMS named ROME4EU [3] hosted on the most powerful device (which is typically the team leader's device). The ROME4EU engine manages the execution of emergency-management processes by orchestrating the human operators with their software applications and some automatic services to access the external data sources and sensors. At the back-end side data sources from several servers are automatically integrated and the result is a single virtual data source that front-end devices can query, thus obtaining information aggregated from several sources. From an organizational perspective, back-end includes the control rooms/headquarters of the diverse organizations that have rescuers involved at front-end. These control rooms provide instructions and information to front-end teams to support their work. Collaboration strictly depends on the possibility that operators and their devices can communicate with each other. Communication is executed on top of ad-hoc networks, such as MANET or Mesh Networks [15, 1], as from Requirement 1 of Section 2. Such mobile networks provide gateways to connect to back-end servers. Figure 2 shows an overview of the front-end WORKPAD infrastructure. The figure refers to one single (front-end) team with different operators who are coordinated in an emergency.

It is worthy observing that WORKPAD deals with the coordination inside the single teams but does not concern the synchronization of activities of processes carried on by different teams. This is a possible continuation of the WORKPAD work; for instance, Franke et al. [6] are being conducted some work on this topic.

For what concerns front-end teams, when a certain task is assigned to a specific operator, the ROME4EU server interacts with the Work-list Handler, which is informed about each assignment made to the respective operator. Both the server and the Handlers are built on top of a Web-service middleware and communicate with each other through web-service endpoints. Each message (e.g., for the notification of a task assignment) is sent through an one-way invocation of a certain method of such endpoints. Once the Work-list Handler Handler receives notification of a certain task assignment to respective users, it displays the name of the task together with relevant information. At any time users can decide to start a task by accepting the offer. In fact, Work-list Handlers do not execute process tasks: tasks are executed with the support of external applications. The Task Handler is only in charge of informing the users about the task assignments and, later, invoking the right applications for task performances.

The development of ROME4EU and, in general, of the entire WORKPAD project has been following an user-centered methodology [8], according to which the development has been carried on in concert with the end users. These have always been confronted with the alternative architectural and implementing choices for the different system releases to evaluate what they considered as best. Such a kind of approach has guaranteed the architecture and the entire system to meet the actual requirements and provide a really efficient and effective system. The goodness of the WORKPAD outcomes is also proven by a showcase held in June, 2009. An earthquake was simulated to occur in the abandoned town of Pentidattilo in southern Italy, and rescue operators were requested to simulate rescue operations with the current means, previously, and, later, by using the entire WORKPAD system, including ROME4EU. From the comparison of the results with and without the system, we have discovered that more efficiency is actually provided with WORKPAD. Moreover, the end users learnt very quickly how to deal with that system, which complies Requirement 5 of Section 2 (see [2] for more details on the analysis with end users).

5 Conclusion

According to our experience and main project outcomes, PAISs are useful during emergency management to improve the effectiveness of first responders and emergency organizations. The process metaphor is generally well understood by end users and the usage of PAISs forced civil protection departments to systemize the procedures to manage emergencies and solve the inherent inefficiencies, thus resulting in a response-time improvement, which is not a direct consequence of the PAIS introduction. Stemming from the experience acquired during the WORKPAD project, in emergency management, information processing and task execution is fully integrated with the physical environment and its objects. The physical interaction with the environment increases the frequency of unexpected contingencies with respect to classical scenarios. So, providing a higher degree of operational flexibility/adaptability is a key requirement of every PAIS for emergency management. From a visualization viewpoint, Task Handlers and other client applications supporting the execution of tasks should be conceived for being used in extreme conditions and under direct sunlight. Therefore, the use of highly-contrasting colors is important (e.g., white on black, yellow on blue). Moreover, users might not have free hands to use PDA's stylus. Therefore, the GUI widgets should be sized in a way that the use of styluses can be avoided (e.g., participants should be able to touch and press buttons by fingers).

References

1. G. Bertelli, M de Leoni, M. Mecella, and J. Dean. Mobile Ad hoc Networks for Collaborative and Mission-critical Mobile Scenarios: a Practical Study. In *WETICE'08: Proceedings of the 17th IEE International Workshops on Enabling Technologies: Infrastructure for collaboration enterprises*. IEEE, 2008.
2. Tiziana Catarci, Massimo Mecella, Massimiliano de Leoni, Andrea Marrella, Manfred Bortenschlager, and Renate Steinmann. The WORKPAD Project Experience: Improving the Disaster Response through Process Management and Geo Collaboration. In *Proceedings of the 7th International Conference on Information Systems for Crisis Response and Management (ISCRAM2010)*, 2010.
3. Massimiliano de Leoni and Massimo Mecella. Mobile process management through web services. In *SCC '10: Proceedings of the 2010 IEEE International Conference on Services Computing*, Washington, DC, USA, 2010. IEEE Computer Society.
4. Massimiliano de Leoni, Massimo Mecella, and Giuseppe De Giacomo. Highly Dynamic Adaptation in Process Management Systems Through Execution Monitoring. In *BPM'07: Proceedings of the 5th Internation Conference on Business Process Management*, volume 4714 of *LNCS*. Springer, 2007.
5. Dirk Fahland and Heiko Woith. Towards process models for disaster response. In *Business Process Management Workshops*, volume 17 of *LNBIP*. Springer, 2009.
6. Jörn Franke, François Charoy, and Cédric Ulmer. A Model for Temporal Coordination of Disaster Response Activities. In *Proceedings of the 7th International Conference on Information Systems for Crisis Response and Management (ISCRAM2010)*, 2010.
7. Dirk Hagebölling and Massimiliano de Leoni. Supporting Emergency Management through Process-Aware Information Systems. In *Business Process Management Workshops*, volume 17 of *Lecture Notes in Business Information Processing*. Springer, 2009.
8. Shah Rukh Humayoun, Tiziana Catarci, Massimiliano de Leoni, Andrea Marrella, Massimo Mecella, Manfred Bortenschlager, and Renate Steinmann. The WORKPAD User Interface and Methodology: Developing Smart and Effective Mobile Applications for Emergency Operators. In *UAHCI '09: Proceedings of the 5th International Conference on Universal Access in Human-Computer Interaction. Part III*, volume 5616 of *LNCS*. Springer, 2009.
9. S. Jul. Who's Really on First? A Domain-Level User and Context Analysis for Response Technology. In B. Van de Walle, P. Burghardt, and C. Nieuwenhuis, editors, *Proceedings of the 4th International Conference on Information Systems for Crisis Response and Management ISCRAM2007*, 2007.
10. Marcello La Rosa and Jan Mendling. Domain-Driven Process Adaptation in Emergency Scenarios). In *Business Process Management Workshops*, volume 17 of *LNBIP*. Springer, 2009.
11. Andreas Lanz, Ulrich Kreher, Manfred Reichert, and Peter Dadam. Enabling Process Support for Advanced Applications with the AristaFlow BPM Suite. In *Proceedings of the Business Process Management 2010 Demonstration Track, September 2010, Hoboken, New Jersey, USA*, volume 615 of *CEUR-WS*, 2010.
12. Leitinger, Sven H. Comparision of GIS-based Public Safety Systems For Emergency Management. In *UDMS '04: Proceedings of 24th Urban Data Management Symposium: Information systems and the delivery of the promised societal benefits*, 2004.
13. Rüdiger Pryss, Julian Tiedeken, and Manfred Reichert. Managing Processes on Mobile Devices: The MARPLE Approach. In *Proceedings of the CAiSE '10 FORUM*, CEUR-WS, 2010.
14. Arthur H.M. ter Hofstede, Wil M.P. van der Aalst, Michael Adams, and Nick Russel. *YAWL and its Support Environment*. Springer, 2009.
15. Murray Turoff. Past and future emergency response information systems. *Communication of the ACM*, 45, April 2002.