

# Using a Workflow Management System to Manage Emergency Plans

**Christian Sell**

SAP Research CEC Dresden  
01178 Dresden, Germany  
christian.sell@sap.com

**Iris Braun**

Technische Universität Dresden  
Chair of Computer Networks  
01062 Dresden, Germany  
iris.braun@tu-dresden.de

## ABSTRACT

In the event of a disaster in Germany a so-called executive staff is set up. In support of their work they refer to emergency plans, which describe the chronological order of a set of suitable measures for a dedicated event e.g. an evacuation. These plans only exist in the form of large printed documents. Hence, the technical support for executing emergency plans is very limited. In this paper we present a model for a workflow management system (WfMS) for supporting the modeling, execution and management of emergency plans before and during a disaster. It is based on the idea that emergency plans are similar to business processes and can therefore be modeled as workflows. In contrast to most traditional WfMS, the introduced approach supports unstructured activities and their delegation as well as the management of resources. Furthermore, we analyze drawbacks of the current process for disaster management using emergency plans.

## Keywords

Workflow management system, workflow model, disaster management, emergency plan.

## INTRODUCTION

In Germany a disaster is defined as a critical situation in which the resources of one organization, e.g. a fire or police department, are not sufficient to manage it. In this case a so-called executive staff is set up. Its task is to work on suitable tactics and measures and to delegate them to its operational units. Furthermore, the staff is responsible for the coordination of its operational units and the distribution and management of necessary resources. In support of their work most such staffs have predefined emergency plans for dedicated events, e.g. an evacuation. Current emergency plans usually exist in the form of multi-page printed documents. Each chapter describes one measure which has to be performed or delegated by the staff. Through interviews with several staff units, we ascertained that suitable approaches for supporting the modeling, execution and management of these plans do not exist. Essentially, members of the staff use common tools such as whiteboards, radio or telephone for this. Computer based tools are mostly limited to office applications like Word or Excel. Our research is focused on the development of a workflow management system (WfMS) to support the modeling, execution and management of emergency plans. We claim that current tools are not suitable for that because they lack resource management and delegation functionalities and do not provide sufficient process support to guide the staff during a disaster. The major objective of our work is to support the staff in the execution of emergency plans.

## PROBLEM ANALYSIS

When using multi-page printed documents as a basis for managing disasters the following drawbacks arise:

- *(P1) Restricted situational overview:* Due to the document structure of emergency plans it is difficult for the staff to keep an overview of completed, running and pending measures and therewith of the whole deployment. However, based on a current situational overview the staff can react more promptly to rapidly changing situations and handle them adequately.
- *(P2) No support for resource management:* Before measures in a disaster can be performed suitably, a set of resources has to be requested and provided. For example regarding the measure “secure injured” a sufficient amount of ambulances and busses have to be available. Currently, telephone, radio and fax are the only means which the staff uses to request resources. There is no further technical support.

- (P3) *Lack of flexibility*: Based on the unpredictable events and fast changing environmental conditions that are typical of disasters, emergency plans have to be adapted before and during execution, in order to minimize the semantic distance between the plan and the real process. Currently, there is no suitable tool support for that. Emergency plans have to be adapted by hand.
- (P4) *No support for delegation*: Currently, the delegation of measures is done by telephone and fax. Drawbacks of this approach are, that phone calls and faxing can be time-consuming and can lead to misunderstandings. Moreover, the provided information can not be traced automatically.

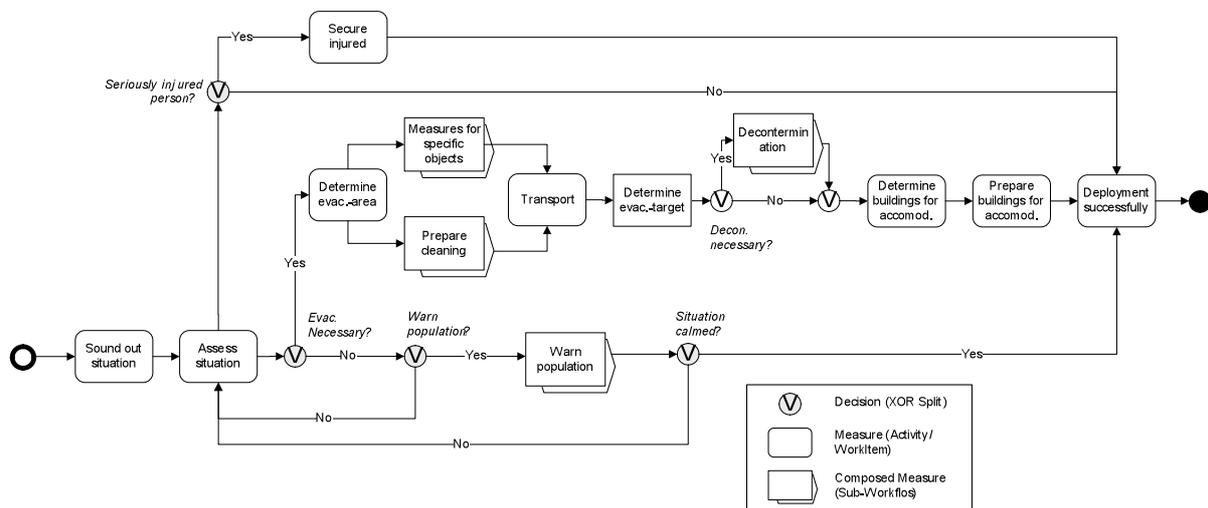
**SOLUTION APPROACH**

**Fundamentals**

The terms *business process*, *workflow* and *workflow management system* are fundamental for our work. The Workflow Management Coalition (Workflow Management Coalition, 1999) defines a business process as “a set of one or more linked procedures or activities which collectively realize a business objective”. A workflow describes the execution of a business process during which activities are passed from one participant to another for action”. Next to the activities, which represent business activities or procedures, workflows consist of control flow elements that determine the order of execution of activities. A WfMS is “a system that defines, creates and manages the execution of workflows through the use of software, running on one workflow engine” (Workflow Management Coalition, 1999).

**Emergency Plan as a Workflow**

Based on a study of several emergency plans and end user interviews we recognized that emergency plans are very similar to business processes. Measures can be seen as a specific form of business activities. As in business processes measures are linked in order to achieve the objective, which is to manage the disaster in the best possible way. This analogy forms the basis of our concept of regarding emergency plans as workflows. Therewith, it is possible to support the modeling, execution and management of emergency plans using a WfMS. Figure 1 shows a part of the emergency plan “Evacuation” modeled as a workflow. It is used in the event of an explosion in a chemical plant when a toxic cloud moves towards a city. At first, the staff has to delegate the measures “Sound out situation” and “Assess situation”. Based on the assessment of the situation, the staff has to decide whether further measures for securing the injured and an evacuation are necessary. In the case of an evacuation, the staff has to start the “actual” evacuation process starting with the measure “Determine evacuation area”. Otherwise the staff has to decide whether the population has to be warned or not.



**Figure 1. Workflow model of an extract from the emergency plan “Evacuation”**

**Basic System Requirements**

Our WfMS has to meet further requirements than “common” WfMS. Those are:

- (R1) *The WfMS must support the management of resources*: This requirement addresses problem P3. The term “management of resources” implies tool support for the whole life cycle of a resource within a disaster. This includes the listing of required resources, the requesting and ordering process, as well as its allocation to an activity.

- (R2) *The WfMS must always depict the current state of the deployment:* To do so the WfMS must display the current state of the workflow, its activities and resources. For example, it has to be visible which activities are being performed or have already been completed.
- (R3) *The WfMS must allow the adaptation of the workflow before and during execution:* This is, e.g., necessary to react to unforeseen events (e.g., an explosion or fast changing weather conditions) which often occur in disasters. The adaptation during execution is required to avoid that time consuming adaptations hinder the execution of further measures. By adaptation we mean the deletion, addition and configuration of activities, connections and resource allocations. Only by adaptation is it possible to keep the emergency plan consistent with the real process.
- (R4) *The WfMS must support the delegation of measures:* The major task of the staff is to determine measures and delegate them to its operational units. Delegation means that the staff hands over the responsibility for the execution and, in certain cases, the resource management to one operational unit.
- (R5) *The WfMS must support the execution of workflows:* Apart from the modeling of emergency plans their execution also has to be supported. Execution means that after the completion of an activity automatically the next pending activity/activities is/are highlighted for the staff. Moreover, in case of alternative routes, the WfMS should support the staff in making the right decision, e.g. based on context information or by highlighting its options.

## WORKFLOW MODEL

### Data Model

As a basis for our WfMS we developed the workflow data model illustrated in Figure 2. The model describes the internal data structure of our workflows. At the same time the data model includes all information necessary to store a workflow and its structural dependencies.

#### *Workflow (emergency plan)*

The `EmergencyWorkflow` is a formal description of an emergency plan. Hence, a plan is represented by an instance of the class `EmergencyWorkflow`. The attributes `progress` and `state` are important to fulfill R3. They will be described in more detail in the next section. Each `EmergencyWorkflow` consists of a set of `ControlFlowElements` and `Activities`.

#### *Activities*

An `Activity` can either be a `WorkItem`, `StartState`, `EndState` or a `SubWorkflow`. A `WorkItem` represents one atomic measure of the staff. Regarding Figure 1, e.g. the measure “Sound out Situation” corresponds to a `WorkItem`. A `WorkItem` can include a `Delegation` and a `ResourceChecklist`. Using the class `Delegation`, the measure is delegated to an operational unit, e.g. via email or instant messaging. Each `WorkItem` can be assigned a number of `Resources` which are necessary to perform it. For this, `Resources` are stored in the `WorkItems` `ResourceChecklist`. In our example, the `ResourceChecklist` of the `WorkItem` “Transport” could include `Resources` like busses, helicopters or cars. Next to storage functionalities the `ResourceChecklist` provides functionalities for requesting and ordering `Resources`. To improve the readability of the `EmergencyWorkflow`, `Activities` can be composed to a `SubWorkflow`.

#### *ControlFlowElements & Connections*

`ControlFlowElements` and `Connections` are used to specify the control flow of the `EmergencyWorkflow`. In this paper we distinguish between `ANDSplits` and `ANDJoins` (for the parallel execution of `WorkItems`/measures) and `XORSplits` and `XORJoins` (for describing alternative paths). Each `XORSplits` is associated with a `Decision`, a staff member has to make. In our example from Figure 1, the staff has to decide after performance of the measure “Assess Situation” if the city has to be evacuated or if the population is supposed to be warned. To link each kind of node `Connections` are used.

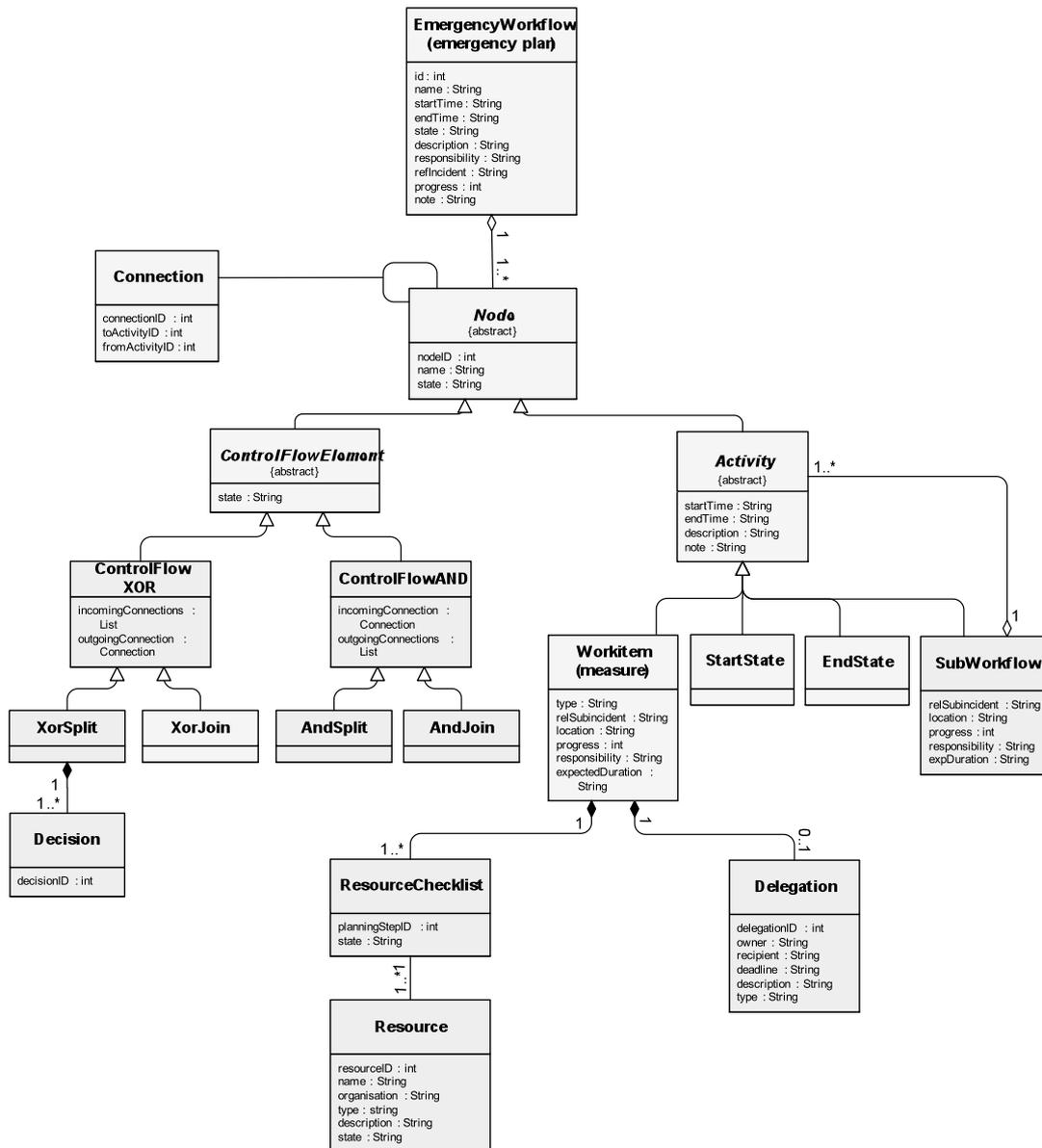


Figure 2. Workflow data model for emergency plans

**Explicit State Management**

To depict the current state of the deployment (R3) the state of each artifact is explicitly modeled. Therefore EmergencyWorkflows, Node, ResourceChecklist and Resources in the data model contain the attribute state. The state of an EmergencyWorkflow can be NOT\_ACTIVATED, ACTIVATED, RUNNING, SUSPENDED, FAILED and COMPLETED. After creation or loading the EmergencyWorkflow is in the state NOT\_ACTIVATED. If a member of the staff is appointed as the responsible person for an EmergencyWorkflow (using the attribute responsibility) it changes to the state ACTIVATED. Now the EmergencyWorkflow can be started. It subsequently enters the state RUNNING. Within this state, the attribute progress shows the progress of the running EmergencyWorkflow. If EmergencyWorkflows are running, they can be SUSPENDED or marked as FAILED. In the case of a successful execution, it is COMPLETED. Figure 3 shows the states of a WorkItem using a UML state chart.

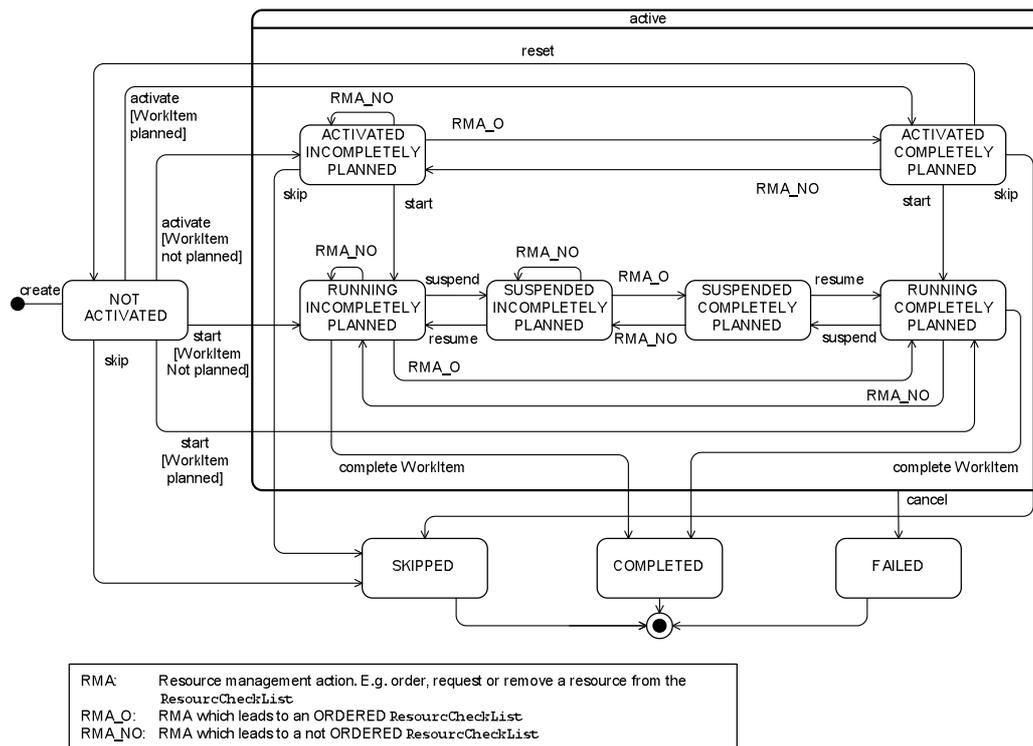


Figure 3. States of a WorkItem

The states NOT\_ACTIVATED, COMPLETED and FAILED correspond to the EmergencyWorkflow states. If a WorkItem does not need to be performed it can be SKIPPED. The other states are composed of the states ACTIVATED, RUNNING, PLANNED, INOMPLETLY\_PLANNED and SUSPENDED. ACTIVATED means that a WorkItem is supposed to be performed but it is still not delegated to an operational unit. After delegating, the staff can mark the WorkItem as RUNNING, meaning, that it is being performed. Furthermore, a WorkItem can be SUSPENDED and can later be resumed. A WorkItem is PLANNED if its ResourceChecklist is in the state COMPLETELY\_ORDERED meaning that all its Resources are successfully ORDERED. Otherwise the WorkItem is UNPLANNED. The states of a ResourceChecklist and a Resource are shown in Figure 4 and 5.

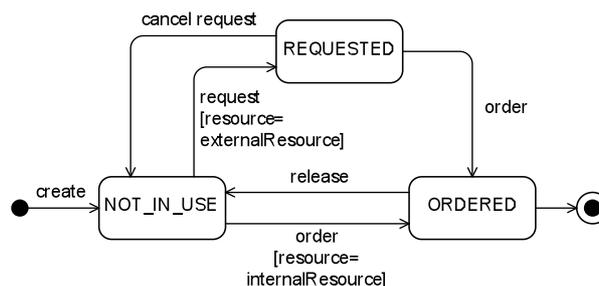


Figure 4. State of a Resource

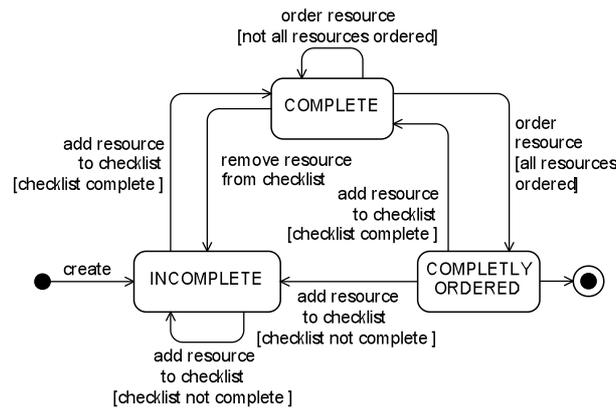


Figure 5. State of a ResourceChecklist

## RELATED WORK

In the literature a lot of workflow models (Buhler and Vidal, 2005; Eder and Gruber, 2002; Li, Gou, Wu and Li, 2005; Lis and Korherr, 2006) and WfMS have been discussed. However, these are not suitable for our concept because they do not explicitly consider resource management, delegation functionalities and state modeling. Furthermore, a number of current WfMS lack adaptation during execution. Some approaches allow for that. However, these are either too restrictive (Narendra, 2000; Whittingham, 1999) or require expert knowledge (Adams, ter Hofstede, Edmond and van der Aalst, 2005; Freßmann, Maximini, Maximini, and Sauer, 2005) that can not be reasonably expected of a staff member.

Current projects which also aim to support the management of emergency plans are ERMA (Peinel, Rose and Berger, 2007) and EUDISMES (Borggräfe, Dörner, Heß, Hofmann, Pipek, Wulf, Scheidl, and Vogel, 2006). Within the project ERMA a “Process Management Module” (PMM) was developed which supports the graphical modeling and management of “risk management processes” which are similar to emergency plans. However, ERMA does not support the execution of them. Moreover, functionalities for resource management and delegation are not provided. In EUDISMES a “Collaborative Task Manager” (CTM) for the modeling and execution of strongly hierarchical and unstructured plans was developed. In contrast to our approach which is based on a process oriented model EUDISMES focuses on a hierarchical model instead. This model does not explicitly regard control-structures for modeling alternatives and parallel branches. This reduces the readability of the plans and therewith the overview of the current deployment.

In (Someren, Netten, Evers, Cramer, de Hoog and Bruinsma, 2006) and (Georgakopoulos, 1999) further approaches to improve disaster management are introduced. The former approach also uses flexible workflows. However, in this case workflows are used to support information dispatching within disasters. In the latter approach a method for process collaboration during a mass casualty is described.

## CONCLUSION AND OUTLOOK

In this paper we presented the idea of using a WfMS to model, execute and manage emergency plans. We analyzed the problems of the current process for disaster management using printed emergency plans and described a set of requirements that a WfMS in the area of disaster management has to meet. Furthermore, we presented a data model which will be the basis for our WfMS and future work. In contrast to common models, it considers the delegation and management of resources. Therewith it addresses the requirements R1 and R4. To better reflect the current state of the deployment (R2), a set of explicitly modeled states is introduced. Requirement R5 is met by the general idea to model emergency plans as workflows and execute them using a WfMS. R3 is not further regarded in this paper and will be part of our future work.

Currently we are working on an architecture for the WfMS which is based on the presented data model. Subsequently, we will implement the first prototype of our WfMS and evaluate it within an empirical study with the fire departments of Cologne and Berlin. The aim of this study is to compare approaches for the execution of emergency plans with and without the use of a WfMS with regard to time and effort reduction.

## REFERENCES

1. Workflow Management Coalition (Feb. 1999) Workflow Management Coalition Terminology & Glossary, Doc.-Nr: WfMC-TC-1011.

*Proceedings of the 6<sup>th</sup> International ISCRAM Conference – Gothenburg, Sweden, May 2009*  
 J. Landgren and S. Jul, eds.

2. Borggräfe, B., Dörner, C., Heß, J., Hofmann, M., Pipek, V., Wulf, V., Scheidl, S. and Vogel, D.T. (2006): EUDISMES – End-User Development in Small and Medium Enterprise Software Systems, in *Statusband Forschungsoffensive Software Engineering 2006*, Federal Ministry of Education and Research, Berlin.
3. Peinel, G., Rose, T. and Berger E. (2007) Process-oriented Risk Management for Smaller Municipalities, *Proceedings 4th International Conference on Information Systems for Crisis Response and Management*, Delft, Netherlands.
4. Eder, J. and Gruber, W. (2002) A Meta Model for Structured Workflows Supporting Workflow Transformations, *Proceedings of the 6th East European Conference on Advances in Databases and Information Systems*, pp. 326-339, London, UK.
5. Li C., Gou J., Wu H. and Li, G. (2005) A process meta-model supporting domain reuse, *2005 International Software Process Workshop*, pp. 459-461, 2005.
6. List, B. and Korherr, B. (2006) An evaluation of conceptual business process modeling languages, *Proceedings of the 2006 ACM symposium on Applied computing*, pp. 1532-1539, New York, NY, USA.
7. Buhler, P. and Vidal, J. M. (2005) Towards Adaptive Workflow Enactment Using Multiagent Systems, *Information Technology and Management Journal*, pp. 61-87, Vol. 6.
8. Whittingham, K. (1999) OpenWater White Paper, IBM Research Division, Zurich Research Laboratory.
9. Narendra, N. C. (2000) Adaptive workflow management - an integrated approach and system architecture, *Proceedings of the 2000 ACM symposium on Applied computing*, pp. 858-865, New York, NY, USA.
10. Adams, M., ter Hofstede, A.H.M., Edmond, D., and van der Aalst, W.M.P. (2005) Facilitating Flexibility and Dynamic Exception Handling in Workflows through Worklets, *Proceedings of 17th Conference on Advanced Information Systems Engineering*, Porto, Portugal.
11. Freßmann, A., Maximini, K., Maximini, R., and Sauer, T. (2005) Collaborative Agent-Based Knowledge Support for Empirical and Knowledge-Intense Processes. *Third German Conference MATES 2005*, Lecture Notes in Computer Science, Vol. 3550., Springer, Koblenz, Germany.
12. Someren, M., Netten, N., Evers, V., Cramer, H., de Hoog, R. and Bruinsma G. (2005) A trainable Information Distribution System to Support Crisis Management, *Proceedings of the 2nd International ISCRAM Conference*, Brussels, Belgium.
13. Georgakopoulos, D. (1999) Collaboration Process Management for Advanced Applications, *International Process Technology Workshop*.