Process Oriented Architecture for Emergency Scenarios in the Czech Republic

Tomáš Ludík, Josef Navrátil, Alena Langerová

Abstract—Tackling emergency situations is performed based on emergency scenarios. These scenarios do not have a uniform form in the Czech Republic. They are unstructured and developed primarily in the text form. This does not allow solving emergency situations efficiently. For this reason, the paper aims at defining a Process Oriented Architecture to support and thus to improve tackling emergency situations in the Czech Republic. The innovative Process Oriented Architecture is based on the Workflow Reference Model while taking into account the options of Business Process Management Suites for the implementation of process oriented emergency scenarios. To verify the proposed architecture the Proof of Concept has been used which covers the reception of an emergency event at the district emergency operations centre. Within the particular implementation of the proposed architecture the Bonita Open Solution has been used. The architecture created in this way is suitable not only for emergency management, but also for educational purposes.

Keywords—Business Process Management Suite, Czech Republic, Emergency Scenarios, Process Execution, Process Oriented Architecture.

I. INTRODUCTION

EMERGENCY Management (EM), sometimes called disaster management, is a discipline of dealing with and avoiding risks. It is a discipline that involves preparing for a disaster before it happens, disaster response, as well as supporting and rebuilding the society after a natural or a human-caused disaster occurs. In general, any EM is a continuous process in which all individuals, groups and communities manage hazards in an effort to avoid or ameliorate the impact of disasters resulting from the hazards [17], [21]. Effective EM relies on an automation of emergency scenarios at all levels of government and non-government participants [5]. For this reason suitable software architecture is necessary.

The software architecture of a system is the set of structures needed to reason about the system, which comprises software elements, relations among them, and properties of both [3]. The term also refers to documentation of a system's software architecture. Documenting software architecture facilitates communication between stakeholders, documents early decisions about high-level design, and allows the reuse of design components and patterns between projects [1].

T. Ludík is with the Department of Population Protection, University of Defence, Brno, Czech Republic (e-mail: tomas.ludik@unob.cz).

J. Navrátil is with the Department of Population Protection, University of Defence, Brno, Czech Republic (e-mail: josef.navratil@unob.cz).

A. Langerová is with the Language Training Centre, University of Defence, Brno, Czech Republic (e-mail: alena.langerova@unob.cz).

There are two types of architectures used for emergency management processes deployment. It is a *Service Oriented Architecture* and *Workflow Management Architecture*. Fundamental differences of these approaches and their use are described below.

A. Service Oriented Architecture

Service-Oriented Architecture (SOA) is an architectural approach that facilitates the creation of loosely coupled, interoperable business services that are easily shared within and among enterprises, SOA derives its true value from the reuse and agility it engenders [19].

In a service-oriented architecture, organizations may use services offered by other companies, and companies may provide services to a growing services market. The vision is for information systems to use business functionality of service providers, so that the reuse of functionality is realized at a level of coarse granularity [27]. New applications can be built with less effort and existing applications can be efficiently adapted to changing requirements, reducing maintenance and development cost.

There are many successful applications of service oriented architecture in EM [10], [11], [26]. For example, the main goal of ORCHESTRA project was to design and implement an open service-oriented software architecture, which improves the interoperability among actors involved in multi-risk management [11]. On the other hand the main goals of the OK-GIS project was the development of a flexible toolbox of software components and services based on a Spatial Data Infrastructure [26].

B. Workflow Management Architecture

Workflow management architecture is based on *Build Time* and *Run Time* approach to process deployment and it is presented by Workflow Management Coalition.

The term Build Time describes a stage during which the computerised form of a business process is generated. During this phase, a business process is translated from the real world into a formal definition, which can be processed by a computer [7]. One or more techniques of analysis, modelling and system design are used. The resulting definition is sometimes called a process model or a process definition.

At Run Time, the process definition is interpreted by software which is responsible for creating and controlling operational instances of the process, scheduling the various activities' steps within the process and invoking the appropriate human and IT application resources [7]. These run-time process control functions act as a link between the

World Academy of Science, Engineering and Technology
International Journal of Economics and Management Engineering
Vol:5, No:11, 2011

process as modelled within the process definition and the process as it is seen in the real world, reflected in the runtime interactions of users and IT application tools.

There are paper dealing with workflow management architecture in EM [14], [20], [22]. MOBIDIS Architecture represents approach, in which operators, equipped with hand-held devices, are coordinated by a workflow management system able to adaptively change the process schema in order to cope with anomalies [14]. In another paper a model for a workflow management system (WfMS) for supporting the modelling, execution and management of emergency plans before and during a disaster is presented [22].

II. STATE OF ART

The Process Oriented Architecture is based on the Workflow Reference Model. It represents a logical view of the Business Process Management Suites which are necessary to automate emergency management scenarios. The chapter also briefly describes the state of the Business Process Management Suite market. The conclusion is devoted to the UML (Unified Modelling Language) and its suitability for describing the new architecture.

A. Workflow Reference Model

The origin of Workflow Management in the present form goes back to the year 1993 when the Workflow Management Coalition (WfMC) was established. WfMC is a global organization of adopters, developers, consultants, analysts, as well as university and research groups engaged in workflow and workflow management. The WfMC creates and contributes to process related standards, educates the market on related issues, and is one of few standards organization that concentrates purely on process.

The WfMC developed an architectural representation of a workflow management system called Workflow Reference Model (Fig. 1). It identifies the most important system interfaces, covering broadly five areas of functionality between a workflow management system and its environment [8].
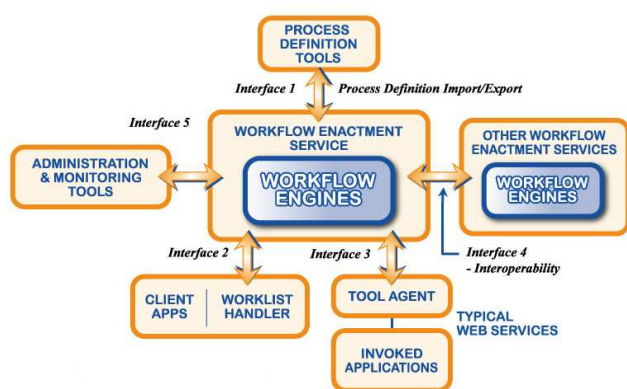

Fig. 1 Workflow Reference Model [7]

B. Business Process Management Suite

For practical deployment of Process Oriented Architecture, software packages called Business Process Management Suite (BPMS) have been used. The BPMS are used to support the entire business process life-cycle, from analysis and modelling through execution to monitoring, the process model must become the core element of the actual business process. Making a model executable requires, besides business process modelling tools, also other BPM enabling software (such as integration technology, a runtime environment and rule engines). Many of the enabling technologies are available separately. When provided together, we call the system the BPMS. The BPMS contains core BPM enabling tools [24]:

Business process modelling tools, also known as business process analysis (BPA) tools, provide a shared environment for the capture, design and simulation of business processes by business analysts, managers, architects and other IT professionals. Models are generally shown in graphical form. BPA tools are modelling-only environments, not execution environments.

Process engines coordinate the sequencing of the activities and steps (system and manual) according to the flows and rules in the process model.

Business intelligence and analysis tools support analysis of data produced during process execution. Capabilities range from reporting to online analytical processing analysis to graphical user dashboards. Business activity monitoring (BAM) systems do this in real time with proactive alerting.

Rule engines execute rules that abstract business policies and decision tables from the underlying applications, and make available more-flexible process changes.

Repositories contain process definitions, process components, process models, business rules and other process data to enable reuse across multiple processes.

Simulation and optimization tools enable business managers to compare new process designs with current operational performance. Scenarios are executed, altering resource constraints and business goals, to assess risk and display the financial and operational (that is, timeliness and quality) impact on the organization.

Integration tools link the model to other system assets (data and logic) that support work steps.

As the market for BPMS matures, additional technologies (such as document management and collaboration support) will be integrated into BPMSs soon.

A BPMS is a set of integrated technologies that enables process stakeholders and users to manage business processes. It includes all the specialty technologies presented above, as well as interfaces to systems for document management, portals, application servers and multiple collaboration technologies. Through the integration of these technologies, the BPMS delivers a consistent process management experience. With help from the IT department, application logic and data can be made accessible to the BPMS, enabling business managers to monitor, analyse and iteratively refine

World Academy of Science, Engineering and Technology
International Journal of Economics and Management Engineering
Vol:5, No:11, 2011

the execution of the process, altering human and automated steps more easily.

C. State of the BPMS Market

Gartner's Magic Quadrant for Business Process Management Suite represents one of several tools that clients can use to evaluate vendors in this market. It also depicts the relative strengths of the top 25 vendors that offer multiregional, cross-industry BPMSs that interest Gartner clients and nonclients the most. These vendors account for most of the spending in the BPMS market. However, clients should also consider other vendors that did not meet our inclusion criteria, such as those specializing in industry-specific processes or in particular geographic locations. The Gartner research is showed in the Fig. 2.

Fig. 2 Gartner's Magic Quadrant for BPMS [24]

IBM Business Process Manager is a comprehensive and consumable BPM platform that provides complete visibility and management of an organization's business processes. It provides a common software platform for process improvement and BPM lifecycle governance, it offers the power and robustness required for mission-critical enterprise solutions, and it combines the simplicity and ease of use required for deeper business engagement [9].

Oracle BPM Suite is the industry's most unified and complete business process management suite. It drives more efficiency, visibility, and agility from a broader range of processes than any other suite of products. Oracle BPM Suite offers the flexibility that the business demands, hand-in-hand with the power IT requires [23].

jBPM is a flexible Business Process Management (BPM) Suite. It makes the bridge between business analysts and developers. Traditional BPM engines have a focus that is

limited to non-technical people only. jBPM has a dual focus: it offers process management features in a way that both business users and developers like it [25].

Bonita Open Solution combines three solutions in one: an innovative Studio for process modelling, a powerful BPM & Workflow engine, and a breakthrough user interface. Create process-based applications in a single day with Bonita Open Solution [2].

The BPMS market is quite diverse. There are many other products such as *Activity*, *Bizagi* or *Tibco Business Studio*. Other tools are displayed in Fig. 2 that shows BPMS basic characteristics in a simple form.

D. Architecture Description with UML

UML, the Unified Modelling Language, is a standard that has wide acceptance and will likely become even more widely used. Although it's original purpose was for detailed design, its ability to describe elements and the relations between them makes it potentially applicable much more broadly. This paper describes our experience using UML to describe the software architecture of a system [6].

For these architecture descriptions, I want a consistent, clear notation that is readily accessible to architects, developers, and managers. It is not mine goal to define a formal architecture description language. The notation could be incomplete, but had to nevertheless capture the most important aspects of the architecture.

Structure diagrams emphasize the things that must be present in the system being modelled. Since structure diagrams represent the structure, they are used extensively in documenting the software architecture of software systems. There are primarily two types of diagrams that should be used for descriptions of software architecture. It is a component diagram and deployment diagram.

Component Diagram illustrates the pieces of software, embedded controllers, etc., that will make up a system. A component diagram has a higher level of abstraction than a Class Diagram - usually a component is implemented by one or more classes (or objects) at runtime. They are building blocks so a component can eventually encompass a large portion of a system.

Deployment Diagram models the run-time architecture of a system. It shows the configuration of the hardware elements (nodes) and shows how software elements and artefacts are mapped onto those nodes. A Node is either a hardware or software element. It is shown as a three-dimensional box shape.

III. PROCESS ORIENTED ARCHITECTURE

The goal of this chapter is to introduce the process architecture design for emergency management. The architecture design is based on workflow reference model, which is introduced in section II.A. Workflow reference model represents the highest logical level of the process architecture, which represents the basic framework for emergency processes deployment. The advantage of this view is that it is

World Academy of Science, Engineering and Technology
International Journal of Economics and Management Engineering
Vol:5, No:11, 2011

independent of specific area of interest and so it can be used for deployment any business processes. The aim of this thesis is to design overall support for deployment emergency management processes. It is also inevitable to add and refill this basic logical design by particular features that are resulted of emergency management processes characteristics, for this reason.

Another starting point in the design process architecture for emergency management was to analyse several BPMS. Its aim is to propose such an architecture that can be implemented using current technologies and resources in the area of process management. Of course, the aim of the proposed architecture is its independence of the specific commercial BPMS solutions. The created architecture represents a fundamental set of tools which can be used to deploy the processes of emergency management. The resulting final solution may consist of one specific BPMS which has already complied with the procedural architecture, or there may be a set of different tools and their interoperability is achieved through standardized interfaces (section III.G), using which the tools can communicate with one another.

The overall process architecture must also be complemented by specific features of emergency management. These result from the study of emergency legislation and other obligatory documents in emergency management. The communication between EM systems occurs in two basic levels. It is a synchronous communication (orchestration) and asynchronous communication (choreography). The synchronous communication is achieved through Invoked Applications while the asynchronous communication is achieved through Other Workflow Enactment Services.

The overall process architecture for emergency management is the result of these considerations (Fig. 3). The architecture is described through the deployment diagram and individual components which the architecture consists of. The detailed description of individual components and their properties is dealt with in detail in next sections.

For the deployment of processes the chosen process architecture created should cover all the necessary tools, modules and communication interfaces among them. A part of the architecture is not only a visual representation of individual components. Components must be described in detail so that the specific BPMS can be chosen according to them.

A. Process Definition Tools

This is the part of architecture responsible for the analysis and design of business processes. For this purpose, primarily three components are used: the *Modelling Tool* for modelling business processes, the *Configuration Tool* for their configuration and integration with other services and also the *Process Repository* which is used for their storage. It also enables the deployment of stored and comprehensively modelled and configured processes to be stored on the process server.

The Modelling Tool represents the editor where you do the development of process applications and toolkits. The editor usually uses a drag-and-drop model for adding activities and other components to the canvas from the palette. The palette is divided into swim lanes which define which participants are authorized to perform which activities. Currently, the BPMN notation is used for modelling processes, which is understandable not only for process analysts, but also for
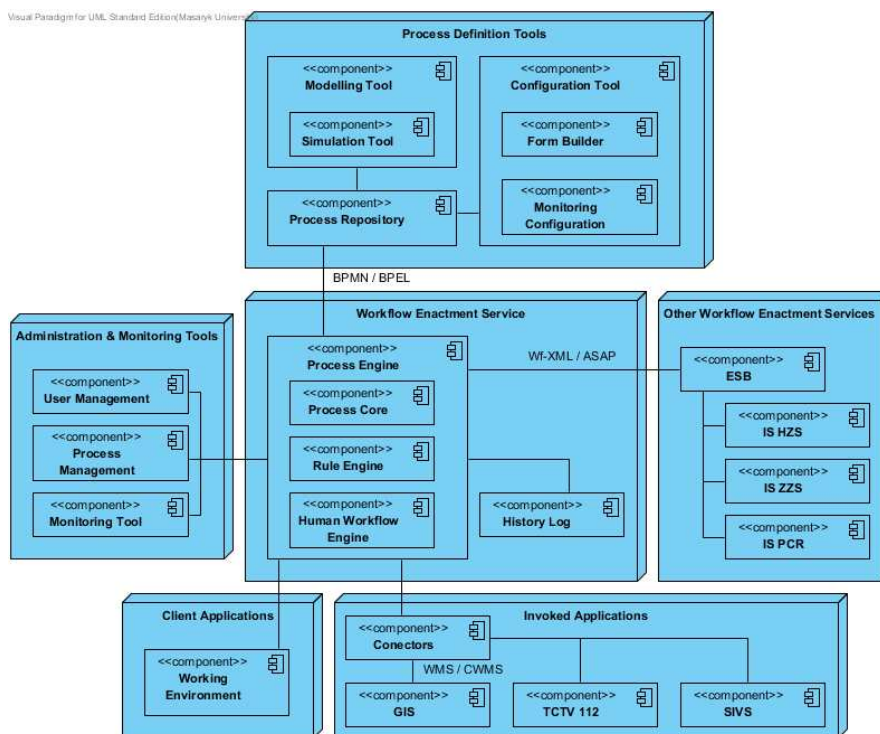

Fig. 3 Process Oriented Architecture for Emergency Scenarios

World Academy of Science, Engineering and Technology
International Journal of Economics and Management Engineering
Vol:5, No:11, 2011

customers and programmers. To verify the accuracy there is a component for process simulation included in the Modelling Tools. On its basis it is possible to evaluate the effectiveness of processes and to eliminate bottlenecks in case of shortcomings in the processes. Through the Simulation Tool a stakeholder can also see the course of process in a virtual environment and thus he/she is able to identify discrepancies between the modelled processes and the reality.

The Configuration Tool is a separate product from the Modelling Tool. However, it is a critical tool used to develop SOA based services that can be called from process applications. Services can be created comprehensively in this tool or it is possible to use the existing services and to map their interfaces in the created process applications. Process applications have a built-in monitoring capability that captures data in the run time. The data can then be used to provide reports regarding the effectiveness of business processes. These tracking and reporting options are often accessible through the built-in Monitoring Configuration module which lets users create the form without advanced knowledge of ICT. The result is that simple processes can be created by normal users. The defined KPIs which reflect long-term objectives set by an organization belong to the basic options of process monitoring. Using these indicators it is possible to comprehensively monitor the entire process. Emergency processes can be divided into two main groups. Processes with human interaction and automated processes. For processes with human interaction it is necessary to create a user interface through which the user interacts with the processes. For this purpose the Form Designer is used which enables creating a transparent form also for the users without an advanced ICT knowledge. The result is that simple processes can be created by common users.

The Process Repository is a tool in which process applications and toolkits are maintained. A process application is a container for process models and their implementations. A process application contains artefacts, such as one or more process models which are called Business Process Definitions (BPD), services required to implement activities or integrate with other systems, Service Component Architecture modules and libraries, toolkits, monitor models and any other items required to run the process. Toolkits are used to provide a shared library of items that are used by multiple process applications.

B. Workflow Enchantment Service

The Process Engine is the core of the process architecture and is required if you want to execute business processes. Your application services typically invoke the Core Engine whenever necessary. The Process Engine is the runtime environment for process applications. It provides a rich and diverse set of functionality for business processes. Business Process Execution Language (BPEL) provides a high level way to define your business processes. Business Process Modelling Notation (BPMN) is another way to define the flow of a business process.

A User task in the process model does not execute task form and actions directly. It instantiates a human task in a separate Human Workflow Service and waits for task completion, instead. At the runtime, end users interact with worklists and task forms through the Human Workflow Service. When a user completes a task, the service reports the completion status back to the process, which resumes at that point.

Business rules let organizations encapsulate business decision logic in reusable components defined outside of the processes that use them. Business rules can be used to simplify complex routing logic at gateways, detailed task assignment and workflow, and dynamic service selection. The business rules are executed and managed in the *Rule Engine*.

An optional core service is the *History Log* that will log all information about the current and previous state of all your process instances. This allows continuous monitoring of running processes and their optimization, if necessary, through business rules or other process setting. Historical data on the state of process instances are also crucial for continuous process improvement using the methods for in-depth analysis of the data by statistical analysis and process mining.

C. Administration and Monitoring Tools

The monitoring system performance enables you to assess performance and evaluate the overall progress of service components that make up the applications deployed on your system. Monitoring the overall performance of the system is essential to understand the performance of application servers, databases, and any other systems critical to your applications. You can also configure BPMS tools to capture the data in a service component at certain event points. This area also includes the settings or modifications of the defined business rule that is possible to edit during the running of process instances and, therefore, modify the runtime behavior of business processes within certain limits. *The Monitoring Tool* component is responsible for the above-mentioned functionality.

The User Management component provides additional options for managing the process from the view of users and their roles. When thinking of user roles, it is important to understand that there is not necessarily a one-to-one correspondence between a person and a role. Every user is typically assigned to one or more user roles that characterize the activities and responsibilities. The Module often provides more comprehensive functionality, such as the substitutability of roles, or its monitoring.

Finally, it is worth to mention the *Process Management* component. Its goal is to provide the basic setup and process configuration (Administration over the entire business process) or individual process activities (Administration over an activity within the business process). The *Administration over the entire business process* means that administrative control over a business process have the authority to terminate, suspend, resume, or delete an instance of that process. On the other hands the *Administration over an activity within the business process* means that administrative control can be

World Academy of Science, Engineering and Technology
International Journal of Economics and Management Engineering
Vol:5, No:11, 2011

granted over invoke, scope, collaboration scope and snippet activities. With the invoked activity, the administrative authority includes the ability to handle faults that may arise, or to force retry or force complete long-running activities. For the collaboration scope and scope activities, the administrative duties include the authority to modify the execution order of the activities nested within the scope activity.

D. Client Applications

The Working Environment represents an integrated user experience for business users across the entire Business Process Management Suite. The Working Environment provides a customizable and collaborative environment for you to monitor, review, and administer common business processes, such as human task flows, modelling, and performance indicators.

The Working Environment is usually a browser-based graphical user interface that you can use to view and interact with content from various components in the BPMS. The Working Environment not only provides a single web-based point of access for the content, you can use the Working Environment to combine the content in useful and interesting ways. These combinations can give you an insight into your business and the capability to react to changes in it.

You can create these combinations by creating mashups. In the Working Environment, a mashup is a set of web applications (widgets) that provide content from multiple sources. The Working Environment depicts these mashups as pages and uses a "working environment" to contain related mashup pages. For example, a working environment could contain widgets from the Monitoring Tool that monitors Key Performance Indicators (KPIs) in your business and widgets from the Process Management module that let users take an action depending on what they see in the KPI widgets. Within a working environment, you can create as many mashup pages as you want to provide multiple views of your business. In addition to viewing content, you can create mashup pages that act on your business. For example, you can create a mashup page to assign people to various tasks or to adjust business rules for different outcomes.

In summary, the Working Environment consists of pages. Each page can contain one or more widgets that are configured to display business information. You can use the widgets to view or interact with that business information. You might be able to access many Working Environments, with each one having a different purpose and, therefore, a different set of pages and widgets.

E. Invoked Applications

Invoked Applications represent such a group of services or applications that communicates with the Process Engine primarily through synchronous communication. A typical example is the communication between the Process Engines and the Geographic Information System (GIS). The standard web services to work with maps called Web Map Service (WMS), or their expansion named Contextual Web Map

Service (CWMS) can be used. CWMS consider not only map transfers, but also other characteristics, such as the visualization for a particular user role in a specific situation.

Communication with the TCTV 112 or SIVS systems can be other representatives of synchronous communication. The TCPV 112 system contains information about reports of incidents at the district operating centre. The SIVS system contains information available through the Czech Hydro-meteorological Institute. This list of systems does not attempt to be comprehensive and only by illustrations it shows the characteristics of the synchronous communication process between the process engine and other services needed for emergency management.

To enable communication between different services and the process engine it is advisable to make Connectors. Connector is an application programming interface (API) that provides an interface between a Process Engine and a legacy application as a dry or SIVS GIS in the field of Emergency Management.

F. Other Workflow Enchantment Services

Other Workflow Enchantment Services represent the primary asynchronous communication. Examples of such interaction are communication with the individual systems of the Integrated Rescue System. The Process Engine sends a task to the external IRS systems and waits for processing. After that an external IRS system sends a message with results back to the Process Engine and the message initialize a new process which is responsible for subsequent processing. The Enterprise Service Bus (ESB) is an appropriate software architecture which captures this kind of communication. ESB properties are described in detail in many other papers [4], [9].

G. Interfaces

The last blank part of the Process Oriented Architecture is its interfaces. For the Process Architecture for Emergency Management it is appropriate to use standardized interfaces, developed by organizations, such as *Workflow Management Coalition*, *Object Management Group* and *Organization for the Advancement of Structured Information Standards*.

Another group of interfaces are interfaces for particular systems. A classic example is a standardized Web Map Service (WMS) defined by the *Open Geospatial Consortium*. For the purposes of emergency management it is also possible to use extensions of this service in the form of Context Web Map Service (CWMS), which is an outcome of the research plan entitled Dynamic Geovisualization in Crisis Management [12]. This extension allows the users not only to obtain maps from the GIS server, but also to obtain maps in a context that may be specific to different users or emergency situations.

The last feature of the interfaces used in the BPMS tools is that many manufacturers use their proprietary solution. Proprietary solutions are usually based on standards introduced by the above mentioned companies, but they are often extended and completed by the requirements and needs of a specific BPMS. This feature often necessitates reliance on

World Academy of Science, Engineering and Technology
International Journal of Economics and Management Engineering
Vol:5, No:11, 2011

a specific supplier and also prevents the interoperability among various BPMS. It is necessary to count on these properties when choosing a particular BPMS for the deployment of emergency management processes.

IV. PROOF OF CONCEPT

The practical usage of Process Oriented Architecture for emergency scenarios in the Czech Republic is illustrated by the proof of concept in which the Business Process Management Suite named *Bonita Open Solution (BOS)* [2] has been chosen. The BOS represents the open source solution for process automation and support. Unlike other commercial BPMS, the BOS has some advantages and disadvantages typical for smaller software solutions based on open source technologies. The Processes *Reporting an Emergency Incident* and *Alarm for Fire Protection Unit* have been chosen for the specific use of the Process Oriented Architecture. This architecture helps automation of emergency scenarios or processes in the Czech Republic.

The Bonita Open Platform Solution is primarily based on the three integrated tools built on the Java programming language. It is Bonita Studio, Bonita Workflow Engine and Bonita User Experience. There are also extensive opportunities for integration with other tools using predefined connectors.

The process automation was based not only on Process Oriented Architecture, but also on Process Oriented Methodology [16], that was presented as one of key components of the Process Framework for Emergency Management [15]. The methodology defines a set of basic phases and procedures which are necessary for the proper analysis and configuration of emergency scenarios and their subsequent automation using the BOS. The Defining, Modelling, Configuration, Execution/Monitoring, and Optimizations are the basic phase of methodology [16].

A. Bonita Studio

The Bonita Studio is a modelling tool that allows process design through the Business Process Modelling Notation (BPMN). In addition, besides a simple drawing, tool is also able to design almost a complete implementation process. For this reason, the tool contains the Form Designer and Connectors. The Connectors are software classes written in Java and they implement the interface. They are intended for partial or full automation of processes in the Bonita Open Solution. From the perspective of Process Oriented Architecture for Emergency Management the Bonita Studio Process covers Definition Tools components and allows the process to interact with Invoked Applications.

The first phase of the Process Oriented Methodology is the Identifying phase. The priority was to familiarize with the processes of the Fire and Rescue Service in this phase. For this purpose, the Emergency Scenarios of Fire Protection Unit were used, which generally contains 127 letters describing the methodological procedures for dealing with emergency situations. The above mentioned processes: *Reporting an Emergency Incident* and *Alarm for Fire Protection Unit* have been selected from this set of documents; they are part of the methodology sheets for the Organization of Intervention (Section O). The following process has been described based on the analysis of these methodical letters.

The Operations Centre operator receives a report on the incident. He/she should draw all the necessary information, e.g. the address of incident or character of incident. In case of suspicion concerning the information validity the operator raises a re-query about the person who provided the information. Then the situation is evaluated and the operator selects the unit or units to be alerted. Within a certain time limit, the rescue unit that is alerted has to confirm the alarm and a subsequent trip to the operational centre, or trip can be revoked due to the failure of intervention technique. The Operating Centre monitors the status of the trip of selected units. If necessary, a new rescue unit is selected, when alerted. The process map shows that the process has two direct participants: Operator of Operation Centre and Fire Protection Unit.

Subsequent process maps are created during the Modelling phase. Outcome process diagrams were created gradually over several iterations of design and implementation. The form of some process parts is influenced by the current functional abilities of the Bonita Open Solution. The whole system is composed of three main processes: *Receiving and Processing Information about the Incident* (responsibility at the Operational Centre), *Alarm of Rescue Unit* (responsibility at the Operational Centre) and the *Acceptance of Alarm and Trip* (responsibility at the Fire Protection Unit). To illustrate the
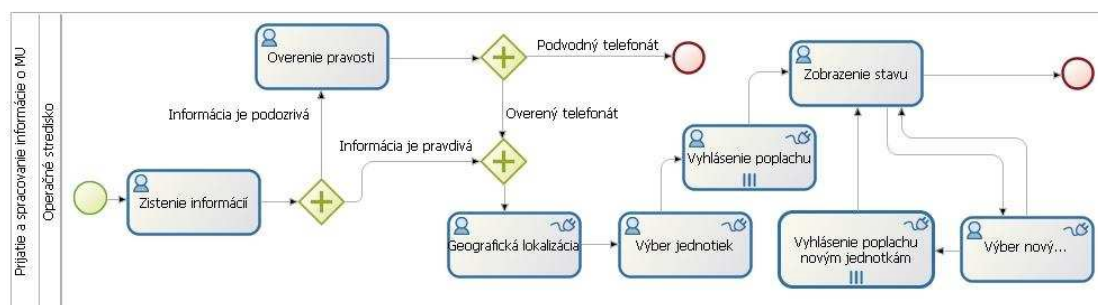

Fig. 4 Process of Receiving and Processing Information about the Incident

World Academy of Science, Engineering and Technology
International Journal of Economics and Management Engineering
Vol:5, No:11, 2011

process model in the Bonita Open Solution process named *Receiving and Processing Information about the Incident* has been selected. The diagram is divided into three parts, which clearly illustrate the main activities on the District Operations Centre site. It is the *Receiving Message about Emergency Incident*, *Evaluation of Information* (including geography) and *Trip Monitoring*, where operations centre monitors all trips of alarmed fire protection units, or sends an additional rescue unit, if necessary.

To ensure the process runtime it is necessary to add forms and connectors to the modelled processes. The Forms and Connectors allow the automation of processes and their operations. Example of a form can be seen in Fig. 5. From such a created process model it is possible to generate a web application that can be deployed on the process server (engine) and that is ready to run.

B. Bonita Workflow Engine

The Bonita Workflow Engine (Bonita Runtime) is a generic and extensible workflow engine which performs proposed process. To achieve the deployment of the entire process on the production server the Bonita Studio Export of applications is necessary. Therefore, a ZIP archive with a complete application is generated. The application contains Bonita Open Solution libraries, Web Console and Client Applications in the WAR8 form of particular processes that can be deployed on the Java EE application server.

Unlike other tools supporting the process runtime, the Bonita Open Solution does not use BPEL to automate the process. The server site application is primarily based on Java which is used for the creation of connectors. The BOS also provides widely used the Groovy language. The Groovy script contains a specialized editor, which facilitates code writing, working with variables, or testing a code. For storing data about Fire Protection Units and data management the MySQL relational database server was used.

C. Bonita User Experience

The Bonita User Experience (XP User) is a Web-based user interface that is similar to web email clients (Gmail) and serves to the entire administration and running processes from both the user and the administrator perspective. The User Experience Bonita accesses workflow engine through client Java API. In the same way, other remote applications can also access. The example of the form which interacts with the user, is showed in Fig. 5.

The Bonita Open Solution contains only simple monitoring tools that are part of the web administration console. The console shows the general indicators of the number of process instances and activities. All these information are preserved even after running the process. These data are not sufficient to fully support the monitoring phase and the subsequent optimization process. Further details as the number of instances of a particular activity, the length of the runtime or utilisation of human resources should be monitored during the monitoring phase.

Practically, it is possible to get additional statistics or raw data necessary for the monitoring phase; there two approaches. The BOS uses the relational database (supports the full range) and data can be obtained by introspection of tables and standard SQL queries. The second approach is to use the API runtime engine. Part of the API is designed to make direct queries about the monitoring process.

The improvement of modelled processes is a long-term activity. For this purpose it is necessary to collect the amount of statistical data and also there are many methods to evaluate these data. The result is a package of recommendations to optimize the current set of processes to achieve even higher performance. The results are often not only more efficient processes, but optimization should also have important impact to the strategic documents of the organization. In this case, they are Emergency Scenarios of the Fire Protection Unit.

Fig. 5 The Form for Reporting an Emergency Incident [18]

D. Integration with Geographic Information System

The Process oriented architecture implementation cannot be considered as complete unless the Bonita Open Solution is integrated with the instruments for cartographic support for emergency management. Particularly, it is contextual cartographic visualization which allows to visualize map elements by a different way for different users in different emergency situations [13]. The outcome imaging device is also very important. It is necessary to display the map on a large screen display and also on a PDA (Personal Digital Assistant) display with a relatively small screen.

Contextual cartographic visualization is achieved by using contextual web map service, which is an extension of the Web Map Service (WMS) specification. WMS is a service designed for sharing geographic data in a distributed internet

World Academy of Science, Engineering and Technology
International Journal of Economics and Management Engineering
Vol:5, No:11, 2011

environment, which makes the geographic information in the maps accessible in various raster image formats. Currently, it is the most used standardized interface for the exchange of maps in digital form on the web.
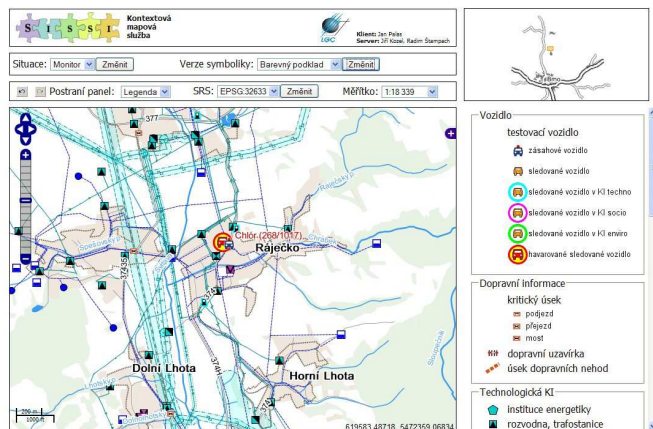

Fig. 6 Contextual Map Client

Therefore, it was necessary to integrate created processes with contextual map client (Fig. 6) allows communication with a context mapping service. The client lets you switch between different contexts, control the displayed map, and also provides a dynamic legend which changes its content depending on the resulting map and its scale.

For practical use of a contextual map client it is important to be able to retrieve the values of parameters passed using the URL. The main information can be obtained through the client such as the Desired Context (context parameter), Centre of the map and Coordinate System (Centre and SRS parameter) and Map Scale (scale denominator parameter).

V. CONCLUSION

The aim of the paper was to define Process Oriented Architecture suitable for the emergency scenario deployment in the Czech Republic. The created architecture defines a set of tools necessary for automation of emergency scenarios and it is independent on the particular BPMS. Bonita Open Solution has been chosen for the validation of this architecture and illustration of its use.

For an overall understanding of the described issue it is also appropriate to familiarize with the Process Framework for Emergency Management [15]. This contribution emphasizes the importance of the two perspectives in the deployment process of emergency management. It should be noted that the architecture itself is not sufficient for the automation of emergency management processes. It should be supplemented by a methodology that defines how to proceed with process automation and deployment. Such methodology has been already published by authors in the article entitled Process Methodology for Emergency Management [16]. This methodology has been used during the automation of selected emergency management scenarios described in this paper.

Finally, it should be noted that the proposed architecture is suitable not only for the automation and deployment of emergency scenarios in the Czech Republic, but also for educational purposes. Thus the defined process oriented architecture allows reflecting any changes in emergency scenarios quickly and also creating scenarios for educational purposes. Students understand emergency planning issues and emergency management more easily when illustrated on real examples. The resulting architecture allows students to model emergency scenarios using process maps, optionally supplemented with additional process data which can enable simulation or deployment on a process engine to test their effectiveness in the model situation in practice.

REFERENCES

[1] L. Bass, P. Clements, R. Kazman, *Software Architecture in Practice*. Second Edition. Boston: Addison-Wesley, 2003. pp. 21–24. ISBN 0321154959.

[2] BonitaSoft, *Bonita Open Solution - User & Reference Guide*. Version 5.5 Rev 1, BonitaSoft, 2011.

[3] P. Clements, F. Bachmann, L. Bass, et al., *Documenting Software Architectures: Views and Beyond*. Second Edition. Boston: Addison-Wesley, 2010. ISBN 0321552687.

[4] M. Fasbinder, BPEL or ESB: Which should you use?, *WebSphere - Technical library*, IBM Corporation, 2008.

[5] J. Fiala, J. Ministr, *Průvodce analýzou modelováním procesů (Process Modelling Analysis Guide)*. Ostrava: VŠB -Technical University, 2003. ISBN 8024805006. (in Czech).

[6] C. Hofmeister, R. L. Nord, D. Soni, Describing Software Architecture with UML. In *Proceedings of the First Working IFIP Conference on Software Architecture*, Kluwer Academic Publishers, 1999.

[7] D. Hollingsworth, *The Workflow Reference Model*. Document Number TC00-1003. Workflow Management Coalition. 1995.

[8] D. Hollingsworth, The Workflow Reference Model: 10 Years On. In *The Workflow Handbook 2004*, 2004.

[9] IBM Corporation, *IBM Business Process Manager*. Thought Leadership White Paper, New York: IBM Corporation, 2011.

[10] M. Ingmarsson, H. Eriksson, N. Hallberg, Exploring Development of Service-Oriented C2 Systems for Emergency Response. In *Proceedings of the 6th International ISCRAM Conference*, Gothenburg, Sweden, 2009.

[11] M. Klopfer, I. Kanellopoulos, *Orchestra, an open service architecture for risk management*. The ORCHESTRA Consorcium. 2008. ISBN 9783000242847.

[12] J. Kozel, *Kontextová mapová služba (Contextual Map Service)*. Dissertation thesis, Brno: Masaryk University, 2009. (in Czech).

[13] P. Kubíček, T. Ludík, E. Muličková, et al., Process Support and Adaptive Geovisualisation in Emergency Management. In *Geographic Information and Cartography for Risk and Crisis Management - Towards Better Solutions*. Berlin Heidelberg: Springer-Verlag, 2010. ISBN 9783642034411.

[14] M. Leoni, F. Rosa, M. Mecella, MOBIDIS: A Pervasive Architecture for Emergency Management. In *Proceedings of the 15th IEEE International Workshops on Enabling Technologies: Infrastructure for Collaborative Enterprises*. Manchester, United Kingdom: IEEE Computer Society, 2006. ISBN 0769526233.

[15] T. Ludík, J. Ráček, Process Framework for Emergency Management - Solving of Emergency Situations by Way of Business Processes. In *Proceedings of the 6th International Conference on Software and Database Technologies*. Portugal: SciTePress, 2011. ISBN 9789898425775.

[16] T. Ludík, J. Ráček, Process Methodology for Emergency Management. In *IFIP Advances in Information and Communication Technology*, Heidelberg: Springer, 2011. ISSN 1868-4238.

[17] H. Mak, A. Mallard, T. Bui, et al., Building online crisis management support using workflow systems. In *Decision Support Systems* 25, Elsevier, 1999.

[18] M. Mayer, *Informační podpora procesů krizového řízení (Information Support of Emergency Management Processes)*. Diploma thesis. Brno: Masaryk University, 2010. (in Czech).

[19] Oracle, *Business Process Management, Service-Oriented Architecture, and Web 2.0: Business Transformation or Train Wreck?* An Oracle White Paper, Oracle, 2008.

[20] V. Pavanasam, C. Subramaniam, M. Mulchandani, et al., Knowledge Based Requirement Engineering Framework for Emergency Management System. In *Second International Conference on Computer Engineering and Applications*, Bali Island, 2010. ISBN 9781424460793.

[21] U. Rüppel, A. Wagenknecht, Improving emergency management by formal dynamic process-modelling. In *24th Conference on Information Technology in Construction*, 2007.

[22] Ch. Sell, I. Braun, Using a Workflow Management System to Manage Emergency Plans. In *Proceedings of the 6th International ISCRAM Conference*. Gothenburg, Sweden. 2009. ISBN 9789163347153.

[23] B. Silver, *A First Look at Oracle BPM Suite 11g*. Bruce Silver Associates, Industry Trend Reports, 2010.

[24] J. Sinur, J. B. Hill, *Magic Quadrant for Business Process Management Suites*. Gartner RAS Core Research Note G00205212, Gartner, 2010.

[25] The jBPM team, *jBPM User Guide*. Version 5.1.0, Community Documentation, jboss.org, 2011.

[26] A. Weiser, A. Zipf, Web Service Orchestration of OGC Web Services for Disaster Management. In *Geomatics Solutions for Disaster Management*, Lecture Notes in Geoinformation and Cartography, 2007. ISBN 9783540721062.

[27] M. Weske, *Business Process Management, Concepts, Languages, Architectures*. Berlin Heidelberg: Springer-Verlag. 2007. ISBN 9783540735212.