Using Scenarios for the Identification of Real-World Events in an Event-Based System

Therese Friberg
Universität Paderborn, C.I.K.
friberg@cik.upb.de

Benedikt Birkhäuser
Universität Paderborn, C.I.K.
b.birkhaeuser@cik.upb.de

Jens Pottebaum
Universität Paderborn, C.I.K.
pottebaum@cik.upb.de

Rainer Koch
Universität Paderborn, C.I.K.
r.koch@cik.upb.de

ABSTRACT

This work focuses on the requirements engineering process of an event-based system in the domain of emergency management. The goal is to identify events which occur and have an effect on the actions and decision making during an operation. We outline a case study to apply scenario-based requirements engineering processes to describe and identify events. Under the special circumstances of the case study one important result is the need of integrating multiple sources into the scenario generation activities due to the singular characteristics of many operations.

Keywords

Scenario-based engineering process, user-centered design, sequence diagram, event trace, real-world events, use cases

MOTIVATION

The context of this research is defined by the domain of German fire protection. Currently the commanders have to rely on information provided by subordinated units transmitted by analogue radio, hand-written message forms and fax. Regular tools used by most fire brigades[1] to visualize the operational picture are maps, resource tables and the communication structure as well as section resp. casualty overviews. Especially for large-scale operations these communication and visualization tools are often not sufficient. The staff personnel involved in decision-making lacks a comprehensive view of the operation.

Mobile ICT systems can bridge event recognition and command tasks to increase speed, precision, efficiency and effectiveness of operations. Therefore requirements need to be gathered to select reasonable sensors types, identify specific events, and classify them in a goal oriented way and transfer these results to functional requirements within concrete use cases. This paper focuses on the identification of specific events to point out the as-is situation of the fire brigade. For the purpose of this paper those events are defined as real-world events. They occur within the direct context of the users in contrast to events which only appear in the software system.

One background for the actual research is the European project SHARE[2]. One important finding derived from this project is the fact that the operational picture is highly based on resource information and less on geographical information. Resource management and especially allocation to tasks requests decision support. The tasks are dependent on events as triggers; fire protection is characterized by highly reactive measures. In general the main outcome of this domain and problem analysis is that (a) decision making processes rely on event recognition in the field, (b) communication of recognized events lacks in effective and efficient techniques

[1] The Fire Departement Dortmund has been involved into the research. The structure of the fire brigade of Dortmund and corresponding processes are representative for German cities with large professional and volunteer fire services. Currently a variety of additional IT systems to expand support functionalities is tested.

[2] http://www.ist-share.org

Reviewing Statement: This paper represents work in progress, an issue for discussion, a case study, best practice or other matters of interest and has been reviewed for clarity, relevance and significance.

and (c) the recognition of events is based on high reconnaissance efforts and profound individual experiences, especially concerning complex events.

Decision problems can be described by using the criteria ‘degree of complexity’, ‘degree of interconnectedness’, ‘degree of non-transparency’ and ‘degree of dynamic’ (Putz-Osterloh, 1992; Dörner, 1987). For the domain of fire fighting the characteristic of the mentioned criteria depends on the type of operation. Our research has resulted in the identification of a demand for support in non-routine operations overcharging the available resources as well as leading to non-routine decision situations and decisions. Such operations can be characterized as having a high degree of complexity, a high degree of interconnectedness, a high degree of non-transparency and are highly dynamic. Klein suggested that decision makers faced to solve decision problems with the above mentioned characteristics often do not use rational choice decision strategies (Klein, 2003). Our research proposes to use an event-driven approach to support situation assessment for the described operations (Birkhäuser, Pottebaum & Koch, 2009).

This research has led to the development of the event-driven system approach for the EU FP7 funded project PRONTO. A system will be developed to gather data from different sensors to fuse the information by the use of event recognition and deliver it through a user-friendly intelligent resource management application. Based on the significant importance of events for the development of an event-based system this paper outlines some aspects of event identification within the scenario-based requirements engineering process taking into account the specific characteristics of the analyzed domain.

REQUIREMENTS ENGINEERING RESEARCH

One major background of this work is set by the literature concerning systems requirements engineering and therefore by 'behavior of the system as seen from outside, for example, by the user' (Dorfman, 2000). The requirements engineering process has the challenge to derive important and beneficial requirements concerning an intended system. McGraw points out that involving of the user in the system development is one of the most considerable requirements engineering activities (McGraw & Harbison, 1997).

An important aspect is the distinction between requirements which reflect the needs of customers and users of a system (Kotonya & Sommerville, 1996) and objectives (Pohl, 2007). For the preparation of the requirements specification the problem analysis resp. the requirements elicitation are defined as the initial and partially overlapping work processes. While the requirements elicitation is defined as one part of the requirements definition process aiming to prepare for the requirements analysis (Thayer & Dorfman, 1990), the problem analysis represents an additional activity (Faulk, 2000).

SCENARIO-BASED REQUIREMENTS ENGINEERING

Pohl emphasizes the importance of scenarios in the requirements engineering process (Pohl, 2007). He shows that the interaction between goals and scenarios leads to goal-oriented requirements.

Scenario-based models represent the current situation and highlight the connections between the user, the existing IT-system and the environment. A scenario is a representation of a demonstration case as an extract of the real-world. Sutcliffe defines scenarios as 'facts describing an existing system and its environment including the behavior of agents and sufficient context information to allow discovery and validation of system requirements' (Sutcliffe, 1998).

Rosson & Carroll explicate that the basic argument behind scenario-based methods is that descriptions of people using technology are essential in discussing and analyzing how the technology is reshaping their activities (Rosson & Carroll, 2002).

McGraw (McGraw & Harbison, 1997) emphasizes the use of event traces (i.e. sequence diagrams) as an instrument to identify real-world events especially for scenarios with a strong temporal character. Jacobson confirms the use of scenarios as a basis for generating use cases (Jacobson et al., 1998). Robertson & Robertson introduce an event-driven process based on use cases (Robertson & Robertson, 2006). Within its first stages the scope of work, adjacent systems, business events and their relations are identified. Based on the results the role of the intended product is determined and corresponding use cases are to be identified.

The connection between events and scenarios is pointed out by Sutcliffe (Sutcliffe, 2002). He outlines the event-driven analysis as one bottom-up method which is suitable for behavior and structural models. All these approaches confirm our direction of work and build a valuable base to develop and refine ideas.

CASE STUDY

As mentioned above our research is focused on support within non-routine operations. With regard to their occurrence those operations are characterized by their singularity. This fact has important implications for the scenario generation. Beside the general overview officers achieve within briefing and debriefing officers involved into the operations are only able to describe the scenario from one distinguished point of view. Thus in most cases officers will not be able to extend their experience in terms of redoing the operation again. This leads to a higher demand of conducting syntheses of various sources for the responsible requirements engineers. It also leads to the need of doing the scenario generation from a superior point of view aggregating different reports and interview protocols to generate a holistic description. By fulfilling that approach an all-embracing perspective is achieved.

Within our project the task has been accomplished by utilizing different reports generated after a real flooding event in Dortmund 2008. Those reports have been validated and supplemented by carrying out interviews with officers, which had different important roles during the described incident.

We used simplified sequence diagrams to represent the scenarios. Within the simplified sequence diagrams every object (actors and participants) of the scenario is presented with a lifeline. Activities are shown with oblong bars (see figure 1). The communication between the objects is illustrated with arrows. Every communication starting point is accentuated with a bullet and the target by an arrowhead. Thus no bullet means that the object is not involved into the communication process. In terms of time the actions in the diagrams are ordered chronological and top down. The creation of the scenario diagrams leads to a clear description of the current use of the system. Every interaction and new emerging object can be identified easily.

An event can be seen as a single point in time when something happens (Allen & Hayes, 1985). Following this definition the real-world events in the diagram can be identified by the arrows. For our scenario generation process all identified events are thereupon listed chronologically and described textually in more detail. In doing so it is important to pay attention to the identical labeling of the real-world events in the diagram and in the descriptive text to guarantee a clear correlation.

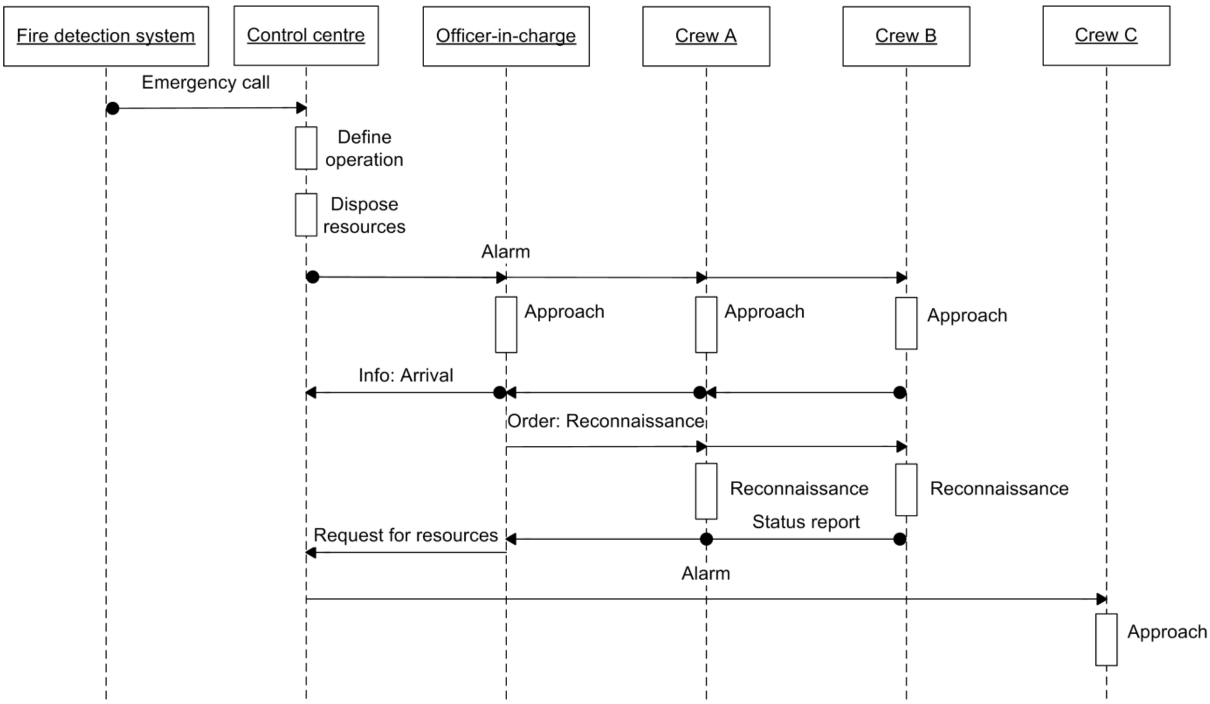

Figure 1: Example of a Sequence Diagram

We applied our method to a complex scenario in the domain of emergency response based on extensive process models. The sequence diagram in figure 1 presents a rough insight into the models we could generate exploring operations of fire brigades in Dortmund. The identified real-world events are described afterwards and as mentioned before the same labeling is used within the diagram and the text.

SCENARIO DESCRIPTION

The context of this research is defined by the domain of German fire protection. The activities performed by the on-site command staff are of special interest. This classification subsumes all commanders on different levels with their assistance. These command and assistance roles are undertaken by officers with specific qualifications and experiences. The initial allocation of personnel to roles is done during the alarming phase by the Operation Control Centre (OCC). After the setup of the command structure these decisions are taken on-site. All officers are given tasks and competencies within a functional, geographical or regulative section. All decisions are based on the cycle of command which is defined in the main directive of German fire brigades (AFW, 1999). On the one hand this cycle is used explicitly for periodic updates of the operational picture. On the other hand officers are trained to utilize the cycle implicitly within their decision making processes. In both cases the planning activities are triggered by events.

In our scenario a fire detection system detects an incident and raises an emergency call. The call is handled by the control centre. This includes the interpretation of the embedded information, the computer aided definition of an operation and disposition of personnel and material. A set of resources which is pre-defined by tactical plans gets alarmed and approaches the site of the incident. The arrival is reported to the control centre. Afterwards the Officer-in-Charge takes over command and sends first orders to the crews. This process is based on the 'cycle of command': Reconnaissance, planning and order. From that point the crew members report to this person instead of to the control centre. Based on reconnaissance results the Officer-in-Charge plans and decides to request additional resources from the control centre. These are alarmed by the regular alerting process and start their approach. In this scenario several aspects are included which can be easily varied to get different sequences of events. For example, the process of starting the operation is much more difficult for the control centre actor getting an emergency call from a human being. Up to this point several real-world events could be identified. They describe specific actions in the presented scenario. For example, the dispositions of resources as well as the arrival of a specific unit at the emergency location are real-world events which can be traced back to the request for additional resources. The request represents another aspect of real-world scenarios, a point in the decision making process.

RESULTS

By using the described techniques to model the scenarios in the context of the fire brigade we succeeded in identifying real-world events which occur during an operation. The different viewpoints of the stakeholders have been aggregated and therefore an overall sight about an incident was achieved. This all-embracing perspective has brought out several real-world events which build the base to identify end-user oriented use cases for the ongoing requirements engineering process. The identified events for the above described example scenario are for example *Defining an operation, Conducting the reconnaissance* and *Request for resources*. The latter is an example for a complex event in real-world: It is based on the event *demand for resources* and corresponding lower level events which can be traced back in the scenarios.

These identified real-world events illustrate the scenario of an operation and covers the occurring events in the environment to which the users might react. The events offer the basis to generate use cases now which contain all emerging events.

CONCLUSION

Our approach is highly dependent on the quality and express ability of the basic scenario. As described we see high potential in building upon existing experiences for requirements analysis and especially use case definition and succeeded for our specific application. For a generic statement on the applicability of our approach to the specific domain we are analyzing the impact of the dynamic incident and environment, the interrelationship with ongoing IT integration and especially alternative decision traces. For the described purpose the applied methodology succeeded in identifying real-world events through an all-embracing perspective. The presented work is part of a software engineering process with strong impact by requirements engineering and evaluation aspects. For the future we will refine our results by following different steps of IT demonstrators throughout the project. They will be used to define requirements and deduct test criteria. Additionally we will extend them from online resp. operational use cases to debriefing and training processes. We will apply our approach to these areas and refine it accordingly.

The research is embedded into an actual system development project leading to a demonstrator system. Throughout the project the results from the requirements engineering phase will be utilized as a basis for system evaluation. Feedback from the involved users let us assume, that the chosen approach is applicable for the targeted purposes and leads to a motivated support.

ACKNOWLEDGMENTS

The authors like to thank the consortium of the PRONTO[3] project, especially the IFR Dortmund and the Fire Department of Dortmund for supporting the requirements engineering process and valuable inputs to this document.

REFERENCES

1. AFW. (1999). *FwDV 100: Führung und Leitung im Einsatz* (Feuerwehr-Dienstvorschrift). Stuttgart.
2. Allen, J. F., & Hayes, P. J. (1985). A common-sense theory of time. In A. Joshi (Ed.), *IJCAI'85: Proceedings of the 9th international joint conference on Artificial intelligence* (pp. 528–531). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
3. Birkhäuser, B., Pottebaum, J., & Koch, R. (2009). Unterstützung von Einsatzentscheidungen der Feuerwehr auf Basis IT-unterstützter Kräftekoordination. In S. Fischer (Ed.), *GI-EditionProceedings: Vol. 154. Informatik 2009. Im Focus das Leben. Beiträge der 39. Jahrestagung der Gesellschaft für Informatik e.V. (GI), 28.9. - 2.10.2009 in Lübeck* (pp. 1393-06). Bonn: Gesellschaft für Informatik.
4. Dorfman, M. (2000). Requirements Engineering. In R. H. Thayer & M. Dorfman (Eds.), *Software requirements engineering* (2nd ed., pp. 7–40). Los Alamitos, Calif.: IEEE Computer Soc. Press.
5. Dörner, D. (1987). *Problemlösen als Informationsverarbeitung* (3rd ed.). Stuttgart: Kohlhammer.
6. Faulk, S. R. (2000). Software Requirements: A Tutorial. In R. H. Thayer & M. Dorfman (Eds.), *Software requirements engineering* (2nd ed., pp. 82–103). Los Alamitos, Calif.: IEEE Computer Soc. Press.
7. Jacobson, I., Christerson, M., Jonsson, P., & Övergaard, G. (1998). *Object-Oriented Software Engineering - A Use Case Driven Approach*. Harlow: Addison-Wesley.
8. James E. Rumbaugh. (1994). Getting Started: Using Use Cases to Capture Requirements. *Journal of Object-Oriented Programming*, *7*(5), 8-12, 23.
9. Klein, G. (2003). *Natürliche Entscheidungsprozesse: Über die "Quellen der Macht", die unsere Entscheidungen lenken*. Paderborn: Junfermann.
10. Kotonya, G., & Sommerville, I. (1996). Requirements engineering with viewpoints. *Software Engineering Journal*, *11*(1), 5–18.
11. McGraw, K. L., & Harbison, K. (1997). *User-centered requirements: The scenario-based engineering process*. Mahwah, NJ: Lawrence Erlbaum Associates.
12. Pohl, K. (2007). *Requirements engineering: Grundlagen, Prinzipien, Techniken* (1. Aufl.). Heidelberg: dpunkt.verl.
13. Putz-Osterloh, W. (1992). Entscheidungsverhalten. In E. Frese (Ed.), *Enzyklopädie der Betriebswirtschaftslehre: Vol. 2. Handwörterbuch der Organisation* (3rd ed., 585–599). Stuttgart: Poeschel.
14. Robertson, S., & Robertson, J. (2006). *Mastering the requirements process* (2nd ed.). Upper Saddle River, NJ: Addison-Wesley.
15. Rosson, M. B., & Carroll, J. M. (2002). *Usability engineering: Scenario-based development of human-computer interaction*. San Francisco, Calif.: Kaufmann.
16. Sutcliffe, A. (1998). Scenario-based requirements analysis. *Requirements Engineering*, *3*(1), 48–65.
17. Sutcliffe, A. (2002). *User-centred requirements engineering*. London: Springer.
18. Thayer, R. H., & Dorfman, M. (1990). *System and software requirements engineering*. Los Alamitos, Calif.: IEEE Computer Society Press.

[3] http://www.ict-pronto.org