Requirements Engineering for Semantic Sensors in Crisis and Disaster Management

Bojan Božić, Mert Gençtürk, Refiz Duro, Yildiray Kabak, Gerald Schimak

Requirements Engineering for Semantic Sensors in Crisis and Disaster Management

Bojan Božić[1], Mert Gençtürk[2], Refiz Duro[1], Yildiray Kabak[2], and Gerald Schimak[1]
[1]Austrian Institute of Technology, Vienna, Austria
[2]Software Research, Development and Consultancy, Ankara, Turkey
bojan.bozic@ait.ac.at

Abstract. This paper describes the requirements engineering methodology used for the definition of semantic sensors in a Crisis and Disaster Management framework. The goal of the framework is effective management of emergencies which depends on timely information availability, reliability and intelligibility. To achieve this, different Command and Control (C2) Systems and Sensor Systems have to cooperate and interoperate. Unless standards and well-defined specifications are used, however, the interoperability of these systems can be very complex. To address this challenge, in the C2-SENSE project, a "profiling" approach will be used to achieve seamless interoperability by addressing all the layers of the communication stack in the security field. The main objective is to develop a profile based Emergency Interoperability framework by the use of existing standards and semantically enriched Web services to expose the functionalities of C2 Systems, Sensor Systems and other Emergency and Crisis Management systems. We introduce the concepts of Semantic Sensors, describe the characteristics of Sensor Systems in Emergency Management, and the methodology of requirements engineering for such a framework.

Keywords: requirements, crisis and disaster management, semantic sensors

1 Introduction

Emergencies and disasters caused by nature and humanity do not recognize country borders and create international problems (see e.g. [1] and [2], and references therein). These include flooding, forest fires, and industrial accidents, to name a few. The Emergency and Disaster Management hence needs cross border solutions, in order to facilitate communication, resource management and offer solutions to emergencies. This is one of the main motivations for the C2-SENSE[1] Framework, where the knowledge, capabilities and experience from a consortium of several European partners[2] is implemented.

---

1 http://c2-sense.eu
2 The list of the C2-SENSE consortium partners is given at http://c2-sense.eu/index.php/partners

adfa, p. 1, 2011.

Due to such a partnership, a specific approach in the framework development is needed, explored and implemented. Therefore, the partners decided to use a dedicated requirements engineering methodology, which is explained in Sec. 4.

The development of an Emergency and Disaster Management tool, such as C2-SENSE, which will apply the current achievements in semantic sensor technologies, is a complex process requiring specific development methods. In our method for the C2-SENSE framework, there exist three different levels of requirements. The first level is the Emergency Domain Inventory, which uses existing standards, real life use cases of sensors, devices, C2 systems and Emergency Management architectures for different scenarios in the security field. The second level is the common Emergency Domain Ontology, which gathers the knowledge of all stakeholders in a unique and flexible data model. The final level is the Emergency Interoperability Profiles which are developed by using the concepts in this ontology and also by taking into account both functional and operational requirements, as well as different countries' cultural, linguistic and legal issues. This is especially important since such profiles include diverse users such as local authority representatives, activists from non-governmental organizations, and police departments.

Standardization activities that aim to evolve the C2-SENSE Emergency Interoperability Framework into a standard specification for interoperability between Sensor Systems and C2 Systems are also part of the requirements engineering process. The framework developed in the C2-SENSE project will assess its outcomes in a realistic pilot that will cover the Puglia region in Italy to ensure that the developed technologies are generic and applicable in a real life setting.

In the following sections, we describe the requirements engineering process in more detail. In Sec. 2, the importance of semantic sensors is discussed. The available and relevant semantic sensor technologies for the C2-SENSE project are presented in Sec. 3. The method used in the requirements engineering process is shown in Sec. 4, while the lessons learned and conclusions are in Sections 5 and 6.

2 Semantic Sensors and Interoperability

To achieve interoperability, sensors and humans need a common ground of communication and understanding, which is expressed in the ontologies.

2.1 Current State of Data and Semantics

Sensor technology is constantly evolving at a high pace; hence we face a constant increase in the amount of new available data. Such a situation asks for an integration of plethora of sensor types that provide diverse capabilities, such as range, modality and maneuverability. Networks with multiple sensors are becoming common ([3], [4]) and can detect and identify objects of interest from great distances and under demanding conditions [5].

There is, however, a lack of proper integration of sensor data leaving us with large amounts of data, but with a low amount of knowledge about this data. For example, an air quality sensor can give us a measurement of the C02 level in the air (a datum), but will probably not provide us with the sensor calibration status or its geospatial location (additional descriptive data). To change this limitation in the field of e.g. the Emergency Management, the sensor data should be annotated with metadata. Technologies such as Semantic Web and Sensor Web Enablement (SWE, [6], [7]) are very suitable for this task. Metadata annotation will help us in order to provide contextual as well as situational awareness information. The basis for this is the concept of spatial, temporal, and thematic metadata for sensors, which is being standardized by the Open Geospatial Consortium (OGC, [8]) and the World Wide Web Consortium (W3C, [9]). Activities related to the extension of the existing standards with Semantic Web technologies have been initiated in both organizations in order to improve the description of and access to sensor data.

2.2 Sensor Interoperability

One of these standardization activities is the development of the OGC Sensor Observation Service (SOS, [10]). This is a Web service specification defined by the OGC's Sensor Web Enablement group in order to standardize the way sensors and sensor data are discovered and accessed on the Web. Besides the main objectives, the standard is providing interoperability between repositories of non-homogeneous sensor data and applications that consume these data. As aforementioned, a lack of knowledge of what kind of environment these data represent is a serious obstacle for data handling applications and value extraction. To solve this, the data need to become smarter, which means that there should exist a more meaningful representation of sensor data. The Semantic Sensor Web initiative models the domain of sensors and sensor observations in a suite of ontologies. Semantic annotations are then added to sensor data by using the ontology models to reason over sensor observations and extend the SOS implementation with semantic knowledge. Such semantically enriched data can be queried by semantically enabled SemSOS [11], i.e. the knowledge of the environment, as well as raw sensor data, can be extracted.

3 Sensor Systems in Emergency Management

An important element in the architecture of a (semantic) sensor network in Emergency Management is related to communication, since the Sensor Systems are based on several protocol messages at different levels. This implies that besides the sensors and hardware, the used communication protocols and standards on different levels are of high relevancy.

For the communication between Sensor Web Services we use Simple Object Access Protocol (SOAP, [12]), which is transported through HTTP, TCP, and IP packets. These are in return sent as Ethernet or WiFi messages. Other kinds of sensors and standards used in the development of C2-SENSE tools are:

- Seamless Communication for Crisis Management (SECRICOM, [13]), which addresses the physical level interoperability for a pervasive and trusted communication infrastructure.
- TErrestrial Trunked Radio (TETRA, [14]), Worldwide Interoperability for Microwave Access (WiMAX, [15]), GSM and WiFi.
- Asset and Resource Management Standards such as Emergency Data Exchange Language - Resource Messaging (EDXL-RM, [16]), Global Justice XML Data Model (GJXDM, [17]), Joint Consultation, Command and Control Information Exchange Data Model (JC3IEDM, [18]), and Hospital AVailability Exchange Language (EDXL-HAVE, [19])
- Notification Management Standards such as OASIS Common Alerting Protocol (EDXL-CAP, [20]), OGC Sensor Web Enablement Information Standards, OMG Alert Management Service (ALMAS, [21]), News Markup Language, or Events Markup Language (EML, [22]).
- Situational Awareness Standards such as EDXL-Situation Reporting (SitRep, [23]), Tracking of Emergency Patients (TEP, [24]), and Emergency Geospatial Data Distribution Standards such as OGC Web Services (OWS, [25]), OGC Keyhole Markup Language (KLM, [26]), OGC Geography Markup Language (GML, [27]) or compact GML.

4 Requirements Engineering Methodology

Well-designed and efficient coordination of ideas and implementations is required when involving several project partners. The requirements engineering methodology implemented in the C2-SENSE framework aims at providing this in a simple and sufficiently flexible way.

4.1 Requirements Engineering Process

The requirements engineering process used in the C2-SENSE project is shown in Fig. 1. Firstly, a state-of-the-art analysis of relevant projects, tools and technologies is prepared. The analysis is used as a primary basis for the second step, i.e. the extraction of C2-SENSE requirements. Further, the requirements in general can be subdivided into system requirements and pilot application requirements. Thus, there is an interchange between these two tasks. Pilot requirements and general requirements tasks cooperate in order to provide a common understanding of the project goals, as well as to synchronize the inputs of users and technical partners who are responsible for the implementation of software components.

Concrete input for requirements engineering is provided by use case diagrams developed by each project partner. The diagrams define the functionality of the respective components, and templates for technical and user requirements, and for the requirements platform database.

The requirements platform database can be accessed via a Web platform interface and is used to categorize different kinds of requirements. It is also used to manage the requirements and their changes during the life-cycle of the framework. Finally, the requirements are used in the architecture and design tasks for defining the architecture of the C2-SENSE system.

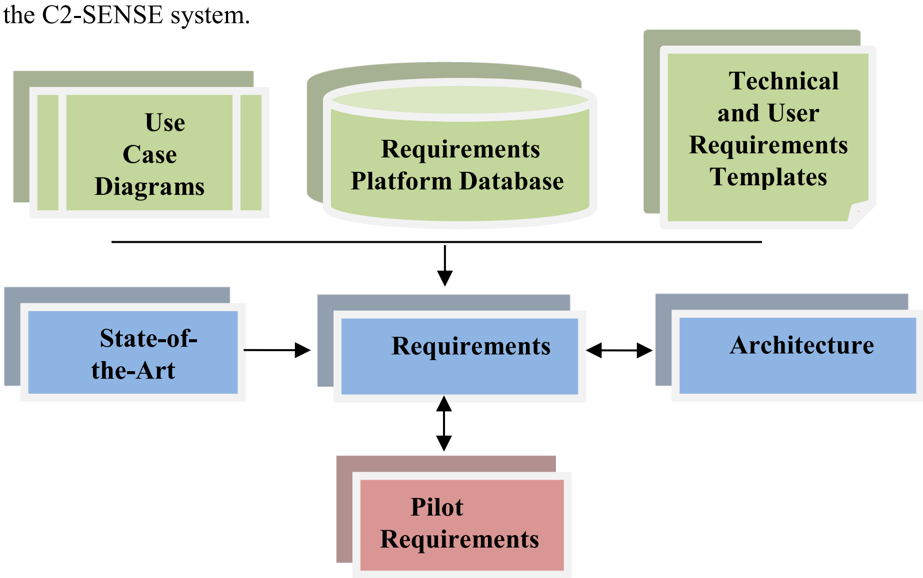

Figure 1: Requirements Engineering Process

In the next section, a more detailed explanation of the engineering process steps is provided.

4.2 Disaster Management Requirements

The following general disaster management requirements are supported by the C2-SENSE platform:

- *Collaborative Decision Making:*
  Collaborative Decision Making supports different users, different command and control systems, and different doctrines and procedures.
- *Tasking:*
  First responders and other units and resources need to be organized, i.e., tasked with assignments.
- *Monitoring and Reporting:*
  Monitoring and Reporting provide position reports and reports on casualties and obstacles.
- *Re-tasking:*
  Re-tasking may be required during a mission to allocate a unit which is available in a given period of time.

4.3 Technical Requirements and Use Cases

C2-SENSE requirements are divided into technical and user requirements. Technical requirements are further subdivided into functional and non-functional requirements.

An example of a functional technical requirement is the one that is related to the interoperability tasks central to the C2-SENSE framework: a development of protocol adapters that provide a standardized, common view of all integrated data sources and services. Such approach produces the framework where the data is linked automatically, as in Linked Data [28].

For non-functional requirements there are notions of independency, scalability, accountability, etc. For example, one of the requirements is that the user interface modules must not rely on proprietary APIs or services that pose the risk of exposing security and privacy sensitive information to third parties. Such requirements are carefully considered in order to make their implementation and usage as smooth as possible.

The user requirements are developed based on use cases that are defined by each partner for their respective component. Moreover, the use cases are directly related to the functional technical requirements, where one technical requirement can have several use cases. A use case describes a specific function required by the user or the user program/system. These are carefully chosen based on the domain user experience (i.e. a fireman, an NGO activist, or a local authority representative), the Emergency and Disaster literature and standards, and the technical requirements. An example of a use case is the "plug and measure" component, i.e. plugging in a sensor that immediately becomes eligible for measurement operations.

When considering responsibilities, each partner is given the possibility to provide the technical and user requirements that fit to that partner's domain and expertise. At the end, the sum of all requirements corresponds to the architecture of the whole system, which means that functional requirements are bound to the system components.

4.4 Templates

In order to collect technical requirements and use cases from all project partners, templates based on [29] and [30] were provided. They were designed as forms and were also used to prepare partner input for the requirements database. The goal of this is to make the cooperation and planning simpler, to identify the possible issues at an early stage, and to ease the transition to the architecture and the implementation.

The technical requirements template consists of a general, but concise description of the C2-SENSE component, a list of use cases implied by the component, a general function description of the component, and a description of the characteristics of the intended component's users, including educational level, experience and/or technical expertise.

The use case template is more extensive. The project partners provide the following information in the template: a description of what the specific use case does and in which context the use case is applied; the scope of the use case (i.e. which system component in the framework is considered); a list of actors involved in the use case, where the actor refers to a person, a role or a system that triggers the use case in order to reach the defined goal; a description of the goal that the actor pursues upon use case execution; a trigger event and the frequency of the use case realization (e.g. once daily, weekly, etc.). Additionally, UML use case and sequence diagrams are provided to explain the use cases in more detail.

4.5 Requirements Platform

The requirements platform is an ancillary way of increasing the efficiency of the requirements engineering process. It is developed to provide a simple method for technical and user requirements input. Moreover, it is meant for the long-term management of requirement changes, as well as for input of new requirements discovered in later phases of the project.

In short, the platform is a web page[3], on which each project partner has a possibility to log and submit the changes to the existing requirements, or to provide new requirements (see Fig. 2). The web platform is structured in a way that the distinctions and relations between the technical and user requirements are very clear. This should increase the efficiency and understanding among the project partners.

Figure 2: Screenshot of the Requirements Database platform of the C2-SENSE framework.

5 Discussion

Our experience with the proposed methodology for requirements elicitation is twofold. At first we started by defining a template in a text document and provided it to project partners based on their responsibilities. The problem, however, was to motivate every partner to participate to the collection of requirements and to show them the importance of having a complete set of framework requirements. The break-

[3] Requirements Database of the FP7 project is located at http://service.ait.ac.at/c2-sense/

through idea was to initiate a web platform for requirements collection and visualize how requirements are related to each other and how the lack of information on a use case can lead to problems in the implementation phase. After understanding the linkage and the possible impacts, in addition to have an easy to use and intuitive platform for collection and management, the process of requirement engineering has improved significantly.

The feedback from all our partners was positive after the change in methodology and we managed to collect user requirements, technical requirements, and use cases in a quick and professional way. It is often very tough to decide when a list of requirements is complete; therefore pitfalls can arise when requirements management is overstressed. On the one hand one does not want to miss important requirements and have an incomplete specification, but on the other hand, putting too much effort in completing the specification phase is not only cumbersome and repetitive for the project team, but also does not lead to a lot of new insights.

Therefore, we recommend a requirement elicitation which supports a dynamic, iterative process and management of new requirements and update of already existing ones throughout the whole project phase.

6 Conclusions

This paper describes the work in the area of requirements engineering specifically applied to semantic sensors in the Emergency Management field. Our contribution is primarily in the proposed method for collection, management, and evolution of requirements, and the development of specific categories for Emergency Management sensor system requirements.

We apply templates that cover the necessary information for technical and user requirements. This is thus more important since there are several international partners involved in the project. A template keeps the ideas concise and simple, which on the other hand increases the understanding in the communication process among partners. We also introduce a web platform, which covers the whole requirements engineering process and which can be reused for similar projects. Our experience with the platform shows that it is effective and efficient, and that it provides a simple way to manage requirements for all the partners involved.

Well-designed and implemented requirements engineering is one of the keys for the success of a project such as C2-SENSE. It is a founding ground for the remaining steps in a process of a system development, and as such is of high importance. It must, however, be simple and flexible enough to allow changes at later stages in the project, and to aid the exchange of ideas among partners. Our proposed method provides the mentioned, is implemented in the C2-SENSE project, and can thus be used in similar projects.

Acknowledgements

The research leading to these results has received funding from the European Community's Seventh Framework Programme (FP7/2007-2013) under grant agreement nr. 607729.

References

[1] Seidel, J. and Imbery, F. and Dostal, P.: Analysis of Historical River Floods - A Contribution Towards Modern Flood Risk Management. INTECH Open Access Publisher, (2012)

[2] Ulbrich, Uwe and Brücher, Tim and Fink, Andreas H. and Leckebusch, Gregor C. and Krüger, Andreas and Pinto, Joaquim G.: The central European oods of August 2002: Part 1 Rainfall periods and ood development. Weather, 371-377, (2003)

[3] Hart, Jane K. and Kirk Martinez: Environmental Sensor Networks: A revolution in the Earth system science? Earth-Science Reviews, 177-191 (2006)

[4] Martinez, Kirk and Hart, K. and Ong, Royan and Brennan, S. and Mielke, A. and Torney, D. and Maccabe, A. and Maroti, M. and Simon, G. and Ledeczi, A. and others: Sensor Network Applications. IEEE Computer, (2004)

[5] Werner-Allen, Geoffrey and Lorincz, Konrad and Welsh, Matt and Marcillo, Omar and Johnson, Jeff and Ruiz, Mario and Lees, Jonathan: Deploying a Wireless Sensor Network on an Active Volcano. IEEE Internet Computing, p. 18-22 (March, 2006)

[6] Jirka, Simon and Br□oring, Arne and Stasch, Christoph: Applying OGC Sensor Web Enablement to Risk Monitoring and Disaster Management. GSDI 11 World Conference, Rotterdam, Netherlands (2009)

[7] Open Geospatial Consortium Sensor Web Enablement (OGC SWE), http://www.opengeospatial.org/ogc/markets-technologies/swe

[8] Open Geospatial Consortim (OGC), www.opengeospatial.org

[9] World Wide Web Consortium (W3C), http://www.w3.org/

[10] Open Geospatial Consortium Sensor Observation Service (OGC SOS), http://www.opengeospatial.org/standards/sos

[11] Henson, C.A. and Pschorr, J.K. and Sheth, A.P. and Thirunarayan, K.: SemSOS: Semantic sensor observation service. IEEE Proceedings of 2009 International Symposium on Collaborative Technologies and Systems, (2009)

[12] Simple Object Access Protocol (SOAP), http://www.w3.org/TR/soap12-part1/

[13] Seamless Communication for Crisis Management Project (SECRICOM), http://www.secricom.eu/

[14] Terrestrial Trunked Radio (TETRA), http://www.tandcca.com/

[15] Worldwide Interoperability for Microwave Access (WiMAX), http://www.wimaxforum.org

[16] Emergency Data Exchange Language Resource Messaging (EDXL-RM), http://docs.oasis-open.org/emergency/edxl-rm/v1.0/errata/EDXL-RM-v1.0-OS-errata-os.pdf

[17] Global Justice XML Data Model (GJXDM), http://tinyurl.com/ndw5gyx
[18] The Joint Consultation, Command and Control Information Exchange Data Model (JC3IEDM), http://tinyurl.com/nr6vgjw
[19] Emergency Data Exchange Language Hospital Availability Exchange (EDXLHAVE), http://docs.oasis-open.org/emergency/edxl-have/v1.0/errata/edxl-havev1.0-os-errata-os.html
[20] Common Alerting Protocol (CAP), http://docs.oasisopen.org/emergency/cap/v1.2/CAP-v1.2-os.html
[21] Alert Management Service (ALMAS), http://www.omg.org/spec/ALMAS/1.0/
[22] Event Markup Language (EML), http://tinyurl.com/nmzz2w7
[23] Emergency Data Exchange Language Situation Reporting (EDXL-SitRep), http://docs.oasis-open.org/emergency/edxl-sitrep/v1.0/edxl-sitrep-v1.0.pdf
[24] Emergency Data Exchange Language Tracking of Emergency Patients (EDXLTEP), http://docs.oasis-open.org/emergency/edxl-tep/v1.0/edxl-tep-v1.0.pdf
[25] Open Geospatial Consortium Web Services (OWS), http://www.opengeospatial.org/standards/owc
[26] Open Geospatial Consortium Keyhole Markup Language (OGC KML), http://www.opengeospatial.org/standards/kml
[27] Open Geospatial Consortium Geography Markup Language (OGC GML), http://www.opengeospatial.org/standards/gml
[28] Bizer, C. and Heath, T. and Berners-Lee, T.: Linked Data - The story so far. Int. J. Semantic Web Inf. Syst., p. 1{22 (2009)
[29] Cockburn, A.: Writing E_ective Use Cases. Addison-Wesley, (2001)
[30] Coleman, D.: A Use Case Template: draft for discussion. Fusion Newsletter, (April, 1998)