Sensor Interoperability for Disaster Management

Harald Rieser, Peter Dorfinger
Salzburg Research Forschungsgesellschaft mbH
Salzburg, Austria
firstname.lastname@salzburgresearch.at

Vangelis Nomikos, Vassilis Papataxiarhis
Department of Informatics and Telecommunications,
Pervasive Computing Research Group,
University of Athens, Greece
{vnomikos, vpap}@di.uoa.gr

***Abstract*—This paper describes how to use sensor information in international disaster management operations. The focus is on enabling sensor interoperability by using standardized interfaces. For this work the Open Geospatial Consortium (OGC) Sensor Observation Service (SOS) is used to exchange sensor information between different systems. Further individual sensor values have to be interpreted to bring benefit to commanders in disaster operations. We are proposing a Sensor Fusion Engine to combine sensor data stemming from heterogeneous sources and provide a condensed output in different standard formats and protocols. An example of such a format is the Common Alerting Protocol (CAP) which is a standardized interface used in disaster operations. Real world deployments in large scale disaster exercises have shown the applicability of the approach.**

Keywords: sensor interoperability, sensor data fusion, disaster management

I. INTRODUCTION

In large scale disaster operations there is often a lack of information. Especially in multinational disasters where foreign units are operating, this lack of information is evident. Even if sensors have been deployed in the area of interest, in most cases this data are expressed in proprietary not standardized formats making their integration impossible. Additionally, sensors usually provide huge amount of data that could not provide intuitive deductions nor can be processed in real-time by humans (even experts). The usage of IT systems could overcome such aspects. The integration of sensor information into such systems and the interpretation of raw sensor data into high-level meaningful information that could be understood by non-experts could be highly beneficial for first responder organizations.

The work presented in this paper is part of a European funded project called IDIRA (Interoperability of data and procedures in large-scale multinational disaster response actions) [1]. The main goal of IDIRA is to improve the cooperation across different responding organizations, by enabling the interoperability between different information systems. Data sources and data consumers are interconnected via standardized interfaces. This leads to more efficient multi-organizational and multinational disaster response actions.

IDIRA starts from the concept that interoperability has to be addressed at both organizational and technical level. This means that involved actors have to align their workflow and procedures first (organizational level). At technical level, the possibility to share data, leading to more efficient cooperation by working on the same set of information, is addressed through the choice of suitable protocols and software interfaces to interconnect information systems, and through the choice of standard data formats to represent and exchange relevant information.

IDIRA provides Inbound and Outbound interoperability software interfaces for data sharing with locally deployed information sources and consumers (like e.g. existing C&C systems). Different standard, non-proprietary data formats are used to exchange different types of information in a structured way: for incidents, resources, or situation reports just to mention a few. The Emergency Data Exchange Language (EDXL) family of standards [7] is used for data about incidents, resources, and for situation reports. The EDXL-CAP (Common Alerting Protocol) [4] data structure is used for Inbound and Outbound incidents data sharing with alerting sources and C&C systems. The EDXL-RM (Resource Messaging) [8] standard allows sharing of data that are relevant for coordinating needs and availability of resources (resource needs and offers, status of resource deployment, and so on). EDXL-SitRep (Situation Reporting) [9] is mainly used to send observations and situation reports through mobile devices, by commanders in the field.

All this information is brought together and visualized in the so called COP (Common Operational Picture). The COP makes it easier for the field commanders to make accurate decisions faster than without the centralized knowledge.

Also sensor information provides valuable information for decision making. For sensor interoperability SOS (Sensor Observation Service) [2] is used. IDIRA provides sensor inbound and sensor outbound interoperability by using SOS. Raw sensor data is integrated into the IDIRA system and transformed to high level concepts through a fusion engine. The output of the fusion engine is formatted and dispatched for visualization in COP. This disaster relevant information can then be taken into account during the decision making process of the commanders in duty.

II. RELATED WORK

The Sensor Web Enablement (SWE) family of standards defined by the Open Geospatial Consortium (OGC) provides a set of standards and protocols that allow the integration of sensor data into spatial data infrastructures. Formats for sensor data and sensor metadata representation are available, as well as service definitions to exchange such data.

Two main components of SWE are relevant for IDIRA. Sensor Observation Service (SOS) [2] is a Web service

Specification for requesting, filtering, and retrieving observations and provides the interface between data processing applications and the sensor data providers.

The Sensor Model Language Encoding Standard (SensorML) [3] specifies models and XML encoding that provide a framework which allows defining the geometric, dynamic, and observational characteristics of sensors and sensor systems. There are many different sensor types, from simple visual thermometers to complex electron microscopes and earth observing satellites. These can all be supported through the definition of atomic process models and process chains. Within SensorML, all processes and components are encoded as application schema of the Feature model in the Geographic Markup Language (GML) Version 3.1.1.

The Common Alerting Protocol (CAP) [4] provides a standardized way to exchange hazard emergency alerts and public warnings over all kinds of media, developed by the OASIS International Open Standards Consortium, using an XML format. The standard describes one alert block with generic incident information, and multiple information parts with detailed, multilingual alert-related information. Each of those blocks may also contain multiple resource blocks for attaching multimedia content, and multiple area blocks for geographic feature information.

In the field of sensor data fusion, several approaches have been proposed to combine data from multiple sensors in order to achieve improved accuracies and more specific inferences compared to the ones resulted by the use of a single sensor alone [17]. In the context of IDIRA, raw data from sensors (temperature, humidity, video, images, etc.) are used in order to improve the accuracy of inference in emergency situations. Such sensors are heterogeneous data sources that call for assessment. Thus, those data streams have to be modified (fused) in order for IDIRA operators to reason about potential emergency situations (e.g., fire). Existing fusion platforms impose a certain (static) processing in the collected data on their route from the data collection layer to the application or focus on different aspects of the Internet of Things (IoT) paradigm (e.g., network support [15]). Even middleware approaches based on Complex Event Processing (CEP), which is a quite recent trend, have quite limiting semantics that are largely dependent upon event occurrence. The proposed architecture is far more versatile than existing platforms [16] as it allows full customization and support of the application domain requirements.

The usage of sensors in disaster management is focusing on a specific purpose. Developed systems are focusing on using a very specific and small subset of sensor types. For example [14] focus on specific sensors and providing this information to first responders, but do not try to provide a generic framework for sensor interoperability in disaster management. To the best of our knowledge this is the first attempt to provide a generic framework for mapping together any kind of sensor information and providing the output as high level information to commanders in disaster operations.

III. The Usage of Sensors in Disaster Management

As described in Section I there is often a lack of information in disaster management. The usage of IT systems and the automatic integration of different data sources could overcome this issue. Sensor information can bring great benefit to the situational awareness of the commanders allowing them better and faster decisions to be taken. A lot of sensor information is existing which could be of potential interest in disaster management. On the one hand mobile sensors which are carried by field units and provide information about toxic gases or similar threats on the other hand static sensors which have already been installed before the actual disaster strikes. Depending on the kind of disaster different sensors like gauging stations, weather stations, fire/smoke detectors, seismographs or sensors embedded in buildings could be integrated. As there is a large number of such sensors existing an integration of raw sensor values would lead to an information overflow for the commanders. There is not only the potential information overflow but also the need to interpret the raw sensor values. As field commanders are not experts in sensor value interpretation, and need not to be, the provided information has to be processed in advance. Within IDIRA we are implementing the concept of sensor fusion to map together different sensor sources, interpret their values all together and provide information to the commanders on a more condensed and higher level view. For instance, when we focus on a flood scenario in Germany, there are different sensor sources available. Services such as Awekas [10], Wunderground [12] and Wedaal [11] can offer sensor measurements from more than 500 weather stations in Germany, or Pegelonline [13] can offer values from more than 500 gauging stations. This will easily sum up to several thousands of individual sensor values integrated into a common endpoint. The above example shows clearly the need for an automatic interpretation of the values. Using a sensor fusion with input data from weather sensors and gauging stations would allow predicting potential floods, without the need of human interpretation of individual sensor values. Consequently the commanders in charge can focus on dealing with handling the disaster itself.

Figure 1 presents the high level architecture of the Sensor subsystem of the IDIRA platform. External sensors as well as sensors under the control of the IDIRA consortium that are deployed on demand provide their values via the Sensor Inbound service of IDIRA. The Sensor Inbound service uses a Message Oriented Middleware to forward all the sensor values to the Sensor Outbound service which provides a single endpoint for making all the sensor information within IDIRA accessible by external systems. In this function IDIRA can be seen as generic hub for sensor data collected in disaster operations. Further the Sensor Inbound service forwards the sensor values to the Sensor Fusion Engine (SFE). The operation of the Sensor Fusion Engine is controlled by the Fusion Expert, a role dedicated to the sensor expert acting at the tactical layer, via the SFE Expert GUI. The Sensor Fusion Engine provides raw sensor values as well as EDXL CAP messages to the Common Operation Picture of IDIRA.

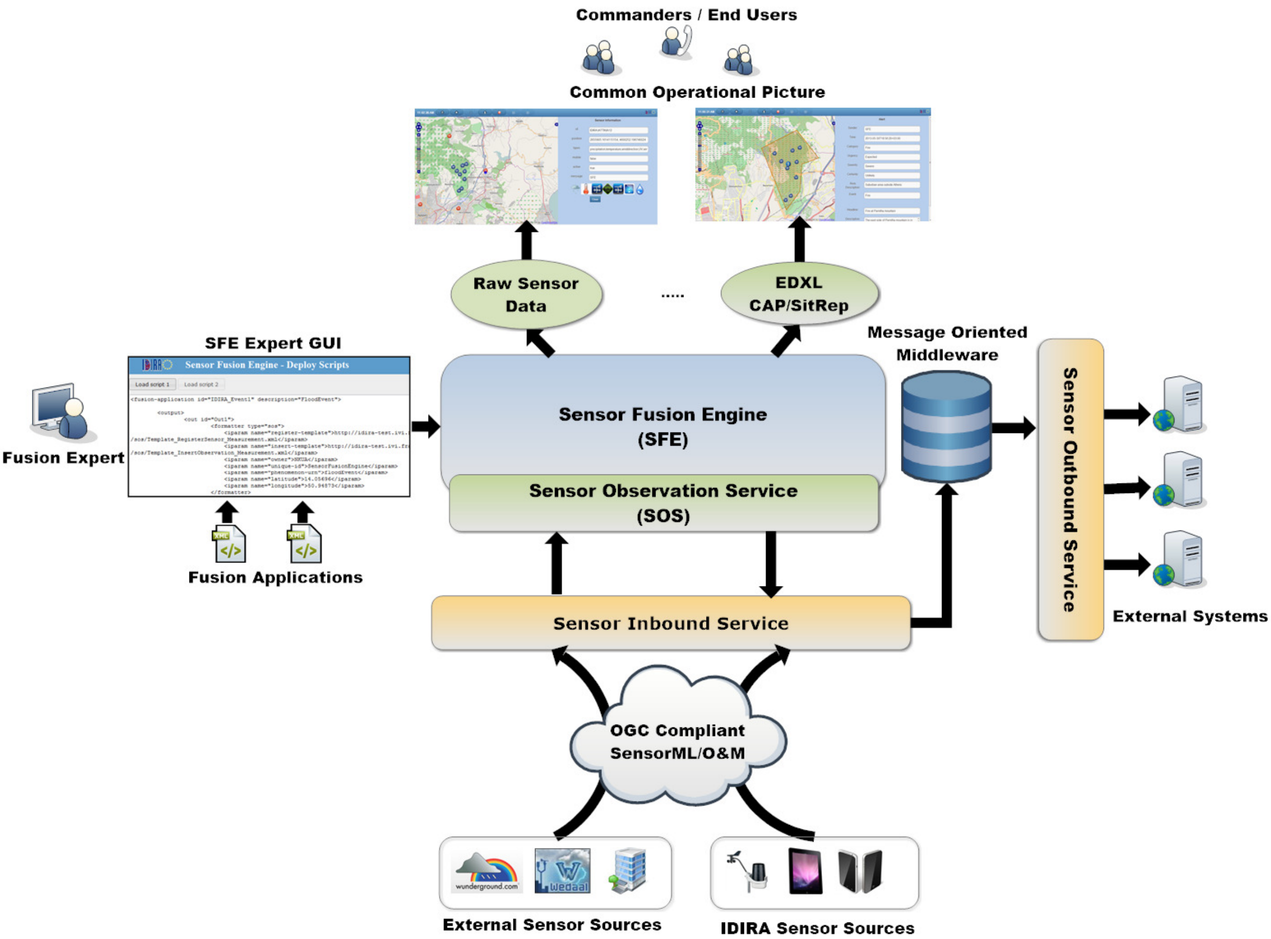

Figure 1: The high level architecture of the Sensor Data Integration subsystem.

IV. Integrated Sensor Sources

For the integration of sensors, the OGC Sensor Observation Service (SOS) has been adopted. This service facilitates the integration of different sensor sources by providing a common, generic interface using open standards. Capabilities of sensors are described using the Sensor Model Language (SensorML), whereas sensor observations and measurements are described using the Observation and Measurement Standard specification (O&M). The service itself describes a process of registering the sensor and sending observations through an HTTP/XML service pattern.

First, an external sensor or sensor system provider has to register itself (as so called *procedure*) with the SOS infrastructure through the Sensor Inbound service, using the *RegisterSensor* operation. This specifies information about the sensor, its location and available sensor values. Once a sensor is successfully registered, the sensor reports observations through the *InsertObservation* operation, making the information available to the SFE and to the Sensor Outbound service.

This approach naturally requires all sensor source to "talk the same language", namely SOS. For most services, this is not the case. For example, Wedaal provides an API using HTTP/JSON, while Wunderground supports a complete REST-API using a proprietary HTTP/XML protocol.

Because of this, even if a standard is used in the inbound/outbound service, a kind of connector/adapter approach is needed to send data to the inbound service, as shown in Figure 2. This approach is used if a sensor provides its data using a non-standard method. Sensor-web enabled sensors may directly connect to the Sensor Inbound service.

A connector connects to an external sensor source using its protocol, requesting updated information. Depending on the source, this can be done using a push (sensor source is able to actively send information to the connector) or pull (polling of the sensor source). The information from the sensor source is then forwarded to the adapter.

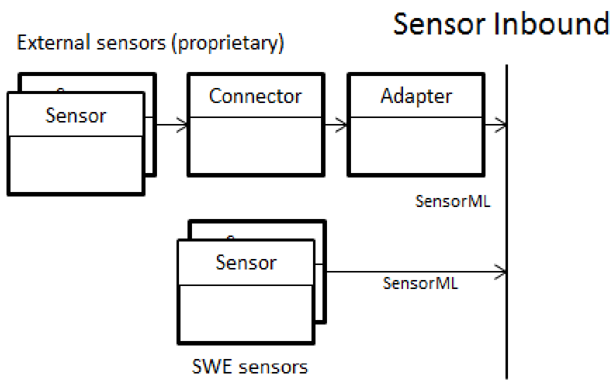

Figure 2: Sensor/Adapter concept.

An adapter has two purposes: converting the message from the proprietary sensor source to SOS/SensorML, and dealing with sensor registration/observation updates. So, it has to perform the following tasks whenever it gets information from the connector: The first step is to check if the sensor sending the information is already registered with SOS. If that is not the case, the sensor is registered using the *RegisterSensor* operation. After this, a check if the observation is new is performed. Finally, the information is sent to the Sensor Inbound Service using the *InsertObservation* operation.

In the IDIRA project, we have adopted this approach for a number of different sensor sources, for example Wedaal, Wunderground and PegelOnline.

Sensors on mobile devices (smartphones and tablets) were also integrated using a similar approach. In the case of mobile devices, an application on the device acts as adapter of local sensor information to SOS/SensorML, sending information such as temperature, magnetic field strength or signal strength of mobile networks to the IDIRA system.

Since sensors on mobile devices are currently of limited use, we also adopted an external mobile sensor-source, namely a Sensordrone-device [5], for use in IDIRA. This device provides information about dangerous gas levels in the environment, and also uses the mobile application as connector/adapter.

V. SENSOR FUSION

The Sensor Fusion Engine (SFE) facilitates the integration and interpretation of different types of sensor data sources, through the definition and execution of multiple algorithmic flows triggered by these sources. The main concept behind this is to produce multiple statistical indices and metrics based on a combination of data coming from different sources, which also, contain some kind of redundancy, in the sense that the same sources can be used to provide different outputs. The use of multiple types of sensor data sources increases the accuracy with which a quantity or phenomenon can be observed, interpreted and used for an event recognition, while redundancy can provide an improved estimate of a physical measurement.

The architecture of the SFE draws its principles from the theory of contextors [6] and leverages the OGC SOS standard for the interoperable integration of sensor data (Figure 3). The core principle of the SFE design is that each contextor encapsulates the functionality provided by a specific algorithm or operator. The SFE provides the necessary middleware services, which will allow contextors to acquire data, execute the encapsulated algorithms, exchange information with each other, and finally produce the necessary output.

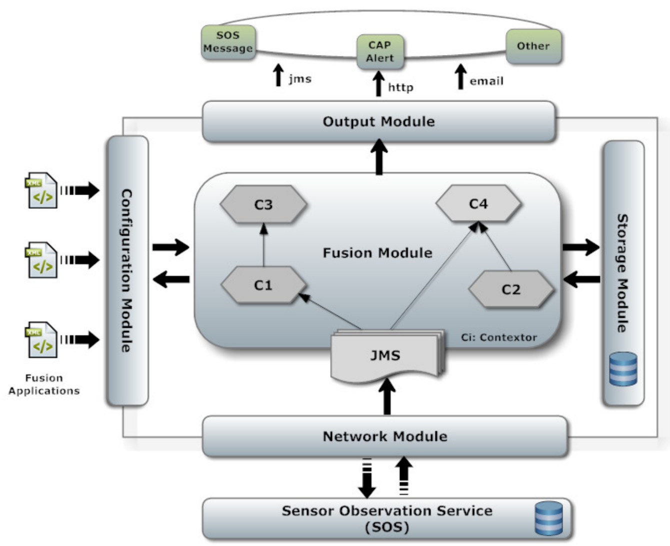

Figure 3: SFE layers.

More specifically, an SOS compliant interface towards the sensor sources conceals the complexities of different underlying communication protocols and data formats, thus allowing for a uniform standardized format for all input streams flowing into the SFE. The Network Module is responsible to recognize the input streams on the fusion process and to collect their values from the underlying database. This selection is based either on spatial restrictions or on restrictions about the measured phenomena and/or specific sensor identifiers. The Fusion Module (FM) is the core component of the SFE in which the fusion process takes place. The structural elements of this component are the contextors and the fusion process is guided through the various interconnections between them. Each contextor encapsulates an algorithm or an operator and is dynamically configured regarding the algorithm's name, parameter(s) and input(s). Depending on the number of the input flows, the provided algorithms from the Fusion Module of the SFE can be categorized as Single Input - Single Output (SISO) and as Multiple Input - Single Output (MISO). In TABLE I the most sophisticated algorithms that the FM provides are presented in a categorization based on the cardinality of input streams. Since the SFE is a stream processing engine that transforms raw sensor data into estimation for an event's occurrence in an area of interest, the produced probability is encapsulated in common message formats that can be easily interpreted by other components. In the context of IDIRA, the most common adopted format for exchanging public warnings and alerts is the Common Alerting Protocol.

As mentioned before, the definition of the information flow for the fusion process must be performed a priori. Through this procedure, the SFE recognizes which sensor streams must be monitored, the contextors that have to be instantiated along with the algorithms that they encapsulate and the output's formatters and adapters that must be exploited. All this information is provided by the deployed fusion application scripts. For the configuration of a fusion application script, an XML based language has been created in order to describe the most significant aspects of the fusion process. This language is known as Application Description Language (ADL). The Fusion Expert can use this meta-language to create scripts, which are next injected in the SFE through the SFE Expert GUI. This tool is web based and provides functionalities for the whole lifecycle of a fusion application script (definition, deployment, monitoring, un-deployment).

TABLE I. ALGORITHMS OF THE SFE

Algorithm	Type	Description
Cumulative Sum Detection Algortihm	SISO	Detects a change on the distribution of a time series $x_t \in \mathbb{R}$ w.r.t. a target value.
Shewhart Control Chart	SISO	A variable x_t is detected to deviate at time k from its normality denoted by two control limits: the Upper Control Limit (UCL) and Lower Control Limit (LCL) [18].
Linear and Symmetric Opinion Pool	MISO	Combines the probabilities of different sources to produce a social probability.
Voting Algorithms	MISO	e.g., threshold voting algorithms [19].
Dempster-Shafer rule of combination	MISO	Part of the Dempster-Shafer theory. Combines evidence from two or more sources to form inferences.
Crisp Value Bayesian Network	MISO	A probabilistic graphical model that represents a set of random variables (with crisp values) and their conditional dependencies [20].

Figure 4 depicts the application definition and deployment functionality of SFE Expert GUI.

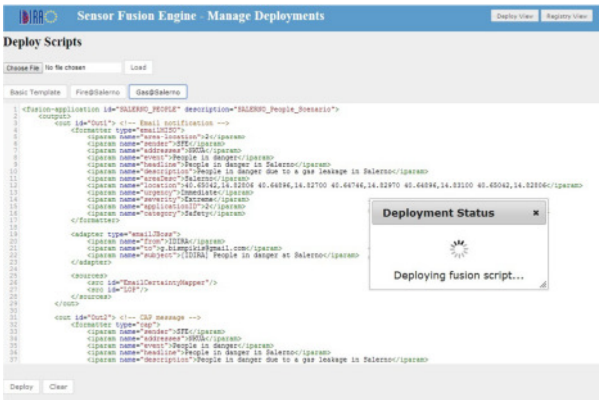

Figure 4: Application definition and deployment functionality of the SFE Expert GUI.

VI. RESULT OUTPUT / REPRESENTATION

In October 2014, a large scale disaster exercise was held in Pirna (Germany) and Görlitz (near Polish border), including forces of Germany, Austria, Poland and Italy (mainly Red Cross, firefighters and THW). During this exercise, parts of the IDIRA system, including the sensor components were used in a real scenario.

The scenario assumed heavy rain over a number of days, causing rivers in the area (Elbe and Neisse) to flood. The IDIRA sensor system was used to predict a flood near the city of Dresden using (simulated) values from gauging stations, public weather data as well as a local weather station. The result of the sensor fusion process was distributed as CAP alerts using the IDIRA system, and was demonstrated on COP presenting an area near Dresden flooded, further initiating steps to prepare evacuation of the predicted area. The result is shown in Figure 5.

This sensor fusion application script used for this scenario has been created in the context of IDIRA for the detection as well as the evolution of a flood incident (i.e., flooded areas) by continuously monitoring the sensor data stemming from the simulated sensors over time. Specifically, the fusion scenario exploited different voting algorithms to fuse multiple sensor streams coming from gauging stations referring to the same topological area. As a second step of the fusion scenario, the result was combined with data stemming from public web services and local weather stations through a linear opinion pool statistical algorithm. The overall output of the fusion process that was defined in this script was the flood probability in the monitored area which was encapsulated in a CAP alert message whenever a threshold was exceeded. The alerting information was subsequently forwarded to the CAP service of IDIRA architecture in order to be presented to actual tactical and field commanders.

Further to the above, it was worth mentioning that sensor measurements, especially for fast reading sensors, may produce very large quantities of measurements. To avoid flooding the COP with useless information, the sensor layer is configured to retrieve and display the last readings for each sensor in more intuitive manners such as diagrams (Figure 6).

Figure 5: Screenshot of the COP application showing sensor sources and the affected area.

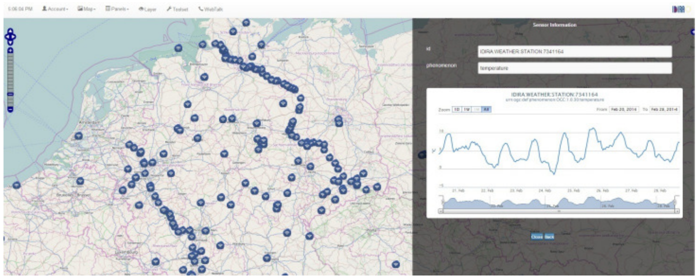

Figure 6: Sensor layer in COP - chart of sensor readings.

In addition to showing data in the COP application, external services and users can access sensor information, and the result of the fusion process, using the outbound services.

Raw sensor data is provided through the Sensor Outbound service. The Sensor Outbound service is an SOS-based implementation, enabling external to query sensor data using the *GetCapabilities* (query general information about the service, as well as sensor metadata), *GetObservation* (query sensor data) and *DescribeSensor* (query metadata about sensors) methods.

Results of the sensor fusion process are presented in CAP format, so any external system able to interpret this format can receive those results. In addition to CAP, it is also possible to send an alerting email to pre-defined email addresses.

VII. Conclusions And Future Work

This paper presents a generic approach for the interoperability of sensor values in disaster operations. Different sensor sources are integrated into a generic platform via an OGC Sensor Observation Service (SOS) standardized interface. All the sensor values from different sources are also provided to the external world via SOS for usage in external system. The heart of the system is a generic Sensor Fusion Engine (SFE) which allows fusion of sensor values from different sources. The configuration of the SFE can be performed via an XML based language. As providing raw sensor values has the potential of an information overflow for the commanders in duty, the SFE interprets different sensor values and performs its output as EDXL-CAP message for visualization purposes. The visualization of the messages in the IDIRA Common Operational Picture (COP) allows taking into account sensor information in the process of disaster management while not being overloaded with detailed information, which are actually not needed in the decision process. To the best of our knowledge this is the only work which allows the integration of generic sensor information and providing a condensed view on the meaning of the values for disaster operations.

Most end users agreed that the sensor integration functionality on COP was very useful during the field trials. The fusion process and the interpretation over raw sensor data provided by the Sensor Fusion Engine through the CAP alert mechanism enabled them to acquire a more accurate picture of the situation in the field. However, additional sensors sources need to be integrated in order to enable the detection of other types of hazardous events

For the future we intend to provide adapters for more sensor sources and implement additional algorithms for the fusion process. Further additional output formats are envisioned. Also, more statistical and machine learning algorithms (e.g., novelty detection, fault detection) are planned to be implemented within the SFE.

Acknowledgment

This work has been partially supported by European Commission within the 7th Framework Programme through project IDIRA (Interoperability of Data and procedures in large-scale multinational Disaster Response Actions), contract 261726, FP7-SEC-2010-1.

References

[1] IDIRA Project, “Interoperability of data and procedures in large-scale multinational disaster response actions”, 2011-2015, http://idira.eu/

[2] Open Geospatial Consortium, Sensor Observation Service (SOS), last accessed 10 June 2014; http://www.opengeospatial.org/standards/sos.

[3] Open Geospatial Consortium, Sensor Markup Language (SensorML), last accessed 10 June 2014; http://www.opengeospatial.org/standards/sensorml

[4] OASIS Emergency Management TC, “Common Alerting Protocol” Version 1.2, 2010-07-01, last accessed 22 October 2014, http://docs.oasis-open.org/emergency/cap/v1.2/CAP-v1.2-os.html

[5] Sensordrone, http://sensorcon.com/products/sensordrone-multisensor-tool

[6] J. Coutaz and G. Rey, “Foundations for a Theory of Contextors”, Computer-Aided Design of User Interfaces (CADUI), Springer, pp. 13-33, May 2002.

[7] OASIS Emergency Management TC, “Emergency Data Exchange Language (EDXL)”, https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=emergency

[8] OASIS Emergency Management TC, “Emergency Data Exchange Language Resource Messaging”, EDXL-RM-v1.0-OS-errata-os, 22 Dec. 2009, https://www.oasis-open.org/standards#edxlrm-v1.0

[9] OASIS Emergency Management TC, “Emergency Data Exchange Language Situation Reporting”, edxl-sitrep-v1.0-wd19, Draft 02, 2012-08-07, http://docs.oasis-open.org/emergency/edxl-sitrep/v1.0/cs01/edxl-sitrep-v1.0-cs01.zip

[10] Othmar Gattringer, “Automatic Weather Map System (AWEKAS)”, http://www.awekas.at (visited Oct 24th 2014)

[11] Wedaal Board “Wedaal” http://www.wedaal.de (visited Oct 24th 2014)

[12] Weather Undergroung, Inc “Weather Underground” http://www.wunderground.com (visited Oct 24th 2014)

[13] Wasser- und Schiffahrsverwaltung des Bundes (WSV) „PEGEL-ONLINE“ http://www.pegelonline.wsv.de (visited Oct 24th 2014)

[14] Mittlboeck, M., Resch, B., Lippautz, M. and Sagl, G. (2010) G2Real: Live Geo(mess)information für Mobile Einsatzkräfte. In: Strobl, J., Blaschke, T., Griesebner, G. (Eds.) (2010) Angewandte Geoinformatik 2010, Wichmann Verlag, Heidelberg, 905-910.

[15] S. Bandyopadhyay, M. Sengupta, S. Maiti and S. Dutta. (2011). “Role of middleware for internet of things: A study”, *International Journal of Computer Science and Engineering Survey*, 2(3), 94-105.

[16] M. Blackstock, N. Kaviani, R. Lea, and A. Friday, “MAGICBroker 2: An open and extensible platform for the Internet of Things,” in *Internet of Things (IOT)*, *2010*, pp. 1–8.

[17] Hall, D.L., & McMullen, S. A. H. (1992). Mathematical Techniques in Multisensor Data Fusion, Artech House. Inc., Norwood, MA.

[18] M. Basseville and I. Nikiforov, “Detection of Abrupt Changes: Theory and Application”, Prentice-Hall, 1993.

[19] B. Parhami, “Voting Algorithms”, IEEE Transactions on Reliability, vol. 43, no. 4, December 1994.

[20] S. Phoha, T. F. La Porta and C. Griffin, “Sensor Network Operations”, Willey-IEEE Press, June 2006.