Improving the Semantic Interoperability of IoT Early Warning Systems: The Port of Valencia Use Case

João Moreira, Luís Ferreira Pires, Marten van Sinderen, Roel Wieringa, Prince Singh, Patrícia Dockhorn Costa and Miguel Llop

Abstract An early warning system (EWS) is a distributed system that monitors the physical world and issues warnings if it detects abnormal situations. The Internet of Things (IoT) offers opportunities to improve monitoring capabilities of EWS and to realize (near) real-time warning and response. This paper presents the development of an interoperable IoT-based EWS to detect accident risks with trucks that deliver goods at the Valencia port area. Our solution addresses the semantic integration of a variety of data sources with processing in safety-critical applications for effective emergency response. The solution considers existing domain-specific ontologies and standards, along with their serialization formats. Accident risks are assessed by monitoring the drivers' vital signs with ECG medical wearables and the trucks' position with speed and accelerometer data. Use cases include the detection of health issues and vehicle collision with dangerous goods. This EWS is developed with the SEMIoTICS framework, which encompasses a model-driven architecture that guides the application of data representations, transformations, and distributed

J. Moreira (✉) · L. Ferreira Pires · M. van Sinderen · R. Wieringa · P. Singh
University of Twente, Enschede, Netherlands
e-mail: j.luizrebelomoreira@utwente.nl

L. Ferreira Pires
e-mail: l.ferreirapires@utwente.nl

M. van Sinderen
e-mail: m.j.vansinderen@utwente.nl

R. Wieringa
e-mail: r.j.wieringa@utwente.nl

P. Singh
e-mail: p.m.singh@utwente.nl

P. D. Costa
Federal University of Espírito Santo (UFES), Vitória, Brazil
e-mail: pdcosta@inf.ufes.br

M. Llop
Valencia Port, Valencia, Spain
e-mail: MLlop@fundacion.valenciaport.com

K. Popplewell et al. (eds.), *Enterprise Interoperability VIII*, Proceedings of the I-ESA Conferences 9, https://doi.org/10.1007/978-3-030-13693-2_2

software components. This framework enables an EWS to act as a semantic broker for situation-aware decision support.

Keywords Semantic interoperability · Early warning system · IoT

1 Introduction

Disaster risk reduction (DRR) is a systematic approach to analyze potential disasters and reduce their occurrence rate and potential impact. The main DRR component is an early warning system (EWS), which is a distributed information system that is able to monitor the physical world and issue warnings if it detects abnormal situations [1]. EWSs can benefit from the Internet of Things (IoT) technologies to realize (near) real-time data acquisition, risk detection, and message brokering between data sources and warnings' destinations [2]. Three major challenges in the development of IoT-based EWS are: (i) semantic integration of a variety of data sources that adhere to different standards, ontologies and data models; (ii) near-real-time processing in time- and safety-critical applications; and (iii) data analysis for effective situation awareness and decision support [2]. In this paper, we describe the SEMIoTICS framework [3], which has been designed to address these challenges. We discuss how SEMIoTICS is being used to develop an interoperable IoT EWS (INTER-IoT-EWS) to detect accidents with trucks delivering goods at the port of Valencia, which is a scenario of the H2020 INTER-IoT project [4]. This project aims to enable semantic integration among IoT platforms at the device, network, middleware, application, and semantic layers. The INTER-IoT-EWS integrates health and logistics data provided by different devices, made available through different IoT platforms and represented with different syntactic and semantic standards. INTER-IoT-EWS use cases include the early detection of a vehicle collision, health issues with drivers, and accidents involving dangerous goods. The use cases' validation plan is presented and lists the performed and current activities. This paper is further structured as: Sect. 2 presents the motivation of our research, Sect. 3 presents the SEMIoTICS framework, Sect. 4 presents the INTER-IoT case study, and Sect. 5 the lessons learned, limitations, and the future work.

2 Motivation

2.1 Early Warning System (EWS)

An EWS is a system for "the provision of timely and effective information, through identified institutions, that allows individuals exposed to a hazard to take action to avoid or reduce their risk and prepare for effective response" [1].

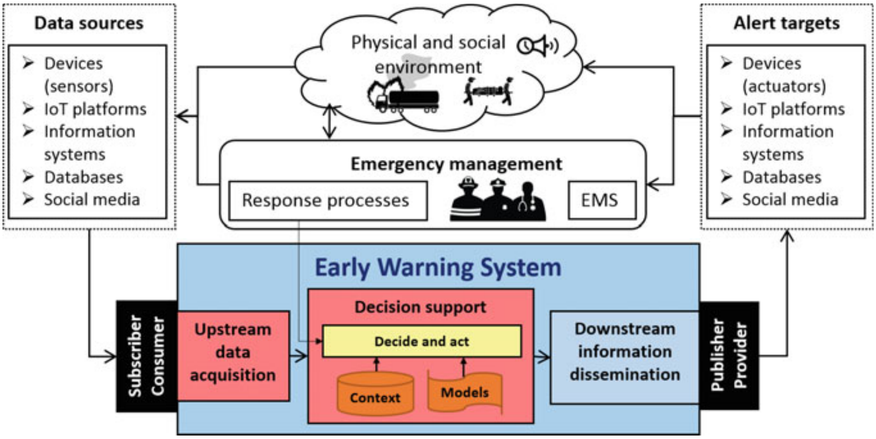

Fig. 1 Typical EWS architecture interacting with the environment

An effective EWS must be people-centered and integrate knowledge about the risks, risks' monitoring and warning, dissemination of meaningful warnings, and public awareness [5]. Modern EWSs comprise software and hardware for data acquisition, situation awareness, decision making, and information dissemination. Some experimental prototypes incorporate IoT technology to improve their functionality [2]. The conceptual architecture of EWS typically consists of three parts [1, 2, 6] (Fig. 1):

- *Upstream data acquisition*: Distributed sensor systems transform observations into digital signals, preprocess the associated data values to ensure that they contain relevant information for decision making, and transmit these data values to a message- and/or event-oriented middleware (broker).
- *Decision support*: Data are stored in data storage and are subject to rules to detect situations of interest. These rules are represented as models, which can be deterministic (e.g., rule-based approach) and/or non-deterministic (e.g., machine learning approach). Once a situation is detected, the EWS considers the requirements of the alert targets to assess the risk and determine the emergency response.
- *Downstream information dissemination*: Different target groups, comprising humans (e.g., the public) and machines (e.g., sirens), receive adequate notifications.

Interoperability is an important feature of effective EWSs for the integration of internal components and interworking of different EWSs. The level of interoperability depends on the standardization of interfaces, data exchange formats, and protocols [6]. The design problem addressed by our research is the improvement of IoT EWSs' interoperability among different data sources and targets, including other EWSs, enabling risky situation identification, and early warning emergency notifications.

2.2 Problem Definition

IoT standards have been defined to improve EWSs syntactic interoperability in multi-agency sensor information integration [7–9], such as the OGC's Sensor Web Enablement (SWE),[1] the OASIS Emergency Data Exchange Language (EDXL)[2], and Health Level Seven (HL7) standards. For example, the FEMA's (USA) Integrated Public Alert and Warning System (IPAWS) and the German Indonesian Tsunami Early Warning System (GITEWS) implement EDXL-CAP, which is a common alert data format protocol [10]. However, these approaches only target syntactic interoperability while we also need semantic interoperability.

The semantic interoperability of EWSs has been addressed by approaches that apply domain-specific ontologies to support meaningful data integration [11, 12]. These semantic solutions usually have poor performance and do not support effective response preparation [13]. In contrast, the *Semantic IoT EWS* approach [2] targets the challenges of scalable time-sensitive data handling from heterogeneous sources, enabling effective responses. This approach balances lightweight and heavyweight semantics: the former for upstream and the latter for downstream data. Moreover, this approach introduces an ontology, the *Decision Support Ontology* (DSO),[3] which is extended with the W3C Semantic Sensor Network (SSN) and OGC SWE terms. Although DSO's goal is "to aggregate and align multiple ontologies to support compound EWS semantics and ontology commitments," it lacks the support for multiple domain ontology alignments at runtime, i.e., it does not provide a mechanism for describing and executing ontology alignments at runtime. Furthermore, the DSO was serialized as XML with the Web Ontology Language (OWL), which inherits the verbosity of RDF/XML and the complexity of OWL, affecting the performance on data exchange and processing.

Our research goal is *to improve the semantic interoperability of emergency services for IoT EWSs, i.e., improve the semantic integration capacity of components of an IoT EWS and enable seamless integration with other IoT EWSs*. We identified the following challenges to achieve this goal:

(C1) *Semantic integration of a variety of data sources:* Avoid loss of semantics when multiple ontologies, standards and data models from different and overlapping domains are involved, considering their syntactic and semantic alignments.

(C2) *Processing in time- and safety-critical applications:* Provide the required performance for upstream data acquisition, emergency risk detection and brokering messages, in terms of scalability and total transaction time.

(C3) *Data analysis for effective responses:* Enable high-quality situation awareness (perception, comprehension, and projection) to avoid false positives and improve decision support based on emergency procedures.

[1] http://www.opengeospatial.org/ogc/markets-technologies/swe.

[2] https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=emergency.

[3] http://tridec.server.de/ontologies/TRIDEC5.2.2.owl#.

3 The SEMIoTICS Framework

The "SEmantic Model-driven development for IoT Interoperability of emergenCy serviceS" (SEMIoTICS) framework aims at improving semantic interoperability among EWSs and their components [3, 14].

SEMIoTICS consists of an architecture (Fig. 2), technologies, and guidelines that are based on model-driven engineering (MDE), inspired by the mediation information system approach [15] and the semantic healthcare lifecycle [16]. SEMIoTICS uses the Endsley's situation awareness theory [17], which is harmonized with the Unified Foundational Ontology [3], enabling high-quality situation awareness (challenge *C3*). The framework also applies MDE transformations to integrate models and generate code for each component of the architecture, according to use case requirements. More details of SEMIoTICS can be found in [18].

The framework architecture has six elements that address the three main functions of an EWS: (1) input handler: upstream data acquisition; (2) abstraction: foundational ontology; (3) context model: domain ontology; (4) situation model: complex event processing; (5) situation awareness: data flows; and (6) output handler: downstream emergency notification. It follows the publisher/subscriber pattern and has RESTful services using JSON and XML, addressing challenge *C1* by enabling web services' syntactic interoperability. JSON for Linked Data (JSON-LD) was adopted in SEMIoTICS to support semantic interoperability and upstream data acquisition performance. JSON-LD is a structured way of using JSON, designed to be a lightweight syntax to serialize RDF, providing interoperability to JSON data at web scale. JSON-LD is a W3C standard recommended by schema.org and Google.

JSON-LD does not fully address challenge *C1* because data can still be represented with multiple different ontologies. To tackle this issue, the architecture supports the identification of functional components that reflect possible decentralized control of EWS functions, recommending interoperability standards to connect these components, and identifying adaptor components to bridge different standards or standards and proprietary solutions. The framework separates adaptors for syntactic interoperability from adaptors for semantic interoperability, allowing adaptor solutions that focus on one particular interoperability problem, and mix and match syntactic and semantic standards with a minimum of different adaptors. Adaptors

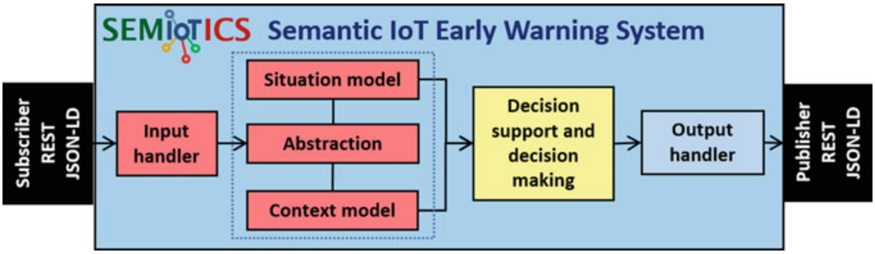

Fig. 2 SEMIoTICS framework for semantic IoT EWS

are implemented as syntactic and semantic translations [19]. The input handler is responsible for message translation, which relies on the syntax of each ontology being used and, therefore, also requires semantic as well as syntactic translations, e.g., from RDF/XML to JSON-LD and from HL7 to EDXL. Messages are translated from the original ontologies to our context model (core ontology) [18], which is aligned to W3C SSN and incorporates terms from EDXL and HL7. This approach aims at facilitating the data and semantics maintenance when integrating distinct domains, measuring, and tracking data quality.

The abstraction component refers to foundational ontologies, which are designed to maximize the support for interoperability of high-level categories, e.g., event, process, physical object, and system. The core ontology and SSN are grounded on the UFO (through OntoUML) and DOLCE Ultralite (DUL), respectively [3]. UFO and DOLCE share the same definitions for some conceptualizations, facilitating the alignment between the ontologies extended with them. This component benefits from the harmonization and axiomatization of terms related to "situation awareness" in UFO [3] and its impact on languages for context and situation modeling, discussed in [18].

The situation model is responsible for the situation identification mechanism, i.e., the formalization of the emergency risk detection [18]. Our approach allows the specification and implementation of complex event processing (CEP). CEP is a common component of IoT platforms to correlate data using temporal predicates (events' relations). For example, Cepheus[4] is the CEP engine of FIWARE IoT platform, based on Java ESPER[5] and, therefore, the event processing language (EPL). The SEMIoTICS guidelines describe how CEP technologies can implement the situation models in ESPER/EPL or SCENE (Drools Fusion) [20]. The situation model addresses *C2*, by incorporating adequate technology, and *C3*, by enabling situation awareness.

Decision support is enabled by the adoption of a workflow management system that enables the end user to design business processes as data flows, e.g., emergency plans. Big data integration tools for workflow development can generate code and are able to deploy data flows at runtime, e.g., Node-RED.[6] We cover the deployment and execution of the data flows for decision making by adopting such tool, addressing challenge *C3*.

The output handler is responsible for brokering the emergency risk notifications to the proper targets, according to the emergency procedures defined in the decision support component. For each predetermined risk, targets are enumerated with their information requirements. The data format of the notifications follows EDXL standards serialized as JSON-LD. Risk notification services are exposed as publishers.

[4] https://catalogue.fiware.org/enablers/iot-data-edge-consolidation-ge-cepheus/.

[5] http://www.espertech.com/.

[6] https://nodered.org/.

4 Case Study: EWS to Detect Accidents at the Port of Valencia

We proposed SEMIoTICS from our research in interoperability of IoT-based emergency systems, and, to validate it, we need to develop an IoT EWS and test in within an emergency scenario (i.e., a case study) composed of use cases. For demonstration and validation purposes, the INTER-IoT project described a scenario to decrease the risk of fatal accidents at the port of Valencia, improving health prevention and enabling quick reaction by reducing time response [21]. The goal is to exploit how e-Health can use IoT platforms dedicated to logistics to prevent the occurrence of accidents and to support evacuation or attention in case of emergency situations.

4.1 Requirements and Use Cases

The requirements of the scenario are:

(FR1) IoT platforms should be able to coordinate with emergency systems by detecting risks of accidents and accidents with trucks within the port area (collision and drivers' health issues), alerting their urgency and severity. The acceptance criterion is to check whether the port IoT platform is able to coordinate with emergency systems located in the vicinity.

(FR2) The hauler IoT platform and the port IoT platform should be able to share health information about the driver, monitored in real time through an electrocardiography (ECG) device. This device should be used for real-time ECG monitoring of drivers, transmitting data to a smartphone, which should act as a gateway, transmitting data to the cloud, both raw and calculated data, e.g., ECG sequence and heart rate (HR). These data need to be integrated with the port emergency control system.

(NFR1) IoT platforms should be semantically and syntactically interoperable. The acceptance criterion is the existence of a mechanism to translate data format and semantics of exchanged message to achieve communication with a common understanding on both sides.

(NFR2) E-Health and logistics should be integrated at the application and semantics level, including primitives for data interpretation of medical and transportation data.

(NFR3) The energy consumption (battery level) of the devices being used for the situation identification mechanism should be monitored.

Five use cases were defined to validate the achievement of these requirements:

(UC01) Vehicle collision detection: Uses accelerometer data of the truck from mobile phone and health device;

(UC02) Hazardous health changes: Detect occurrences of stress and arrhythmia (e.g., bradycardia and tachycardia);

(UC03) Temporal relations between UC01 and UC02: Detect if a health issue occurred before, during, or after a vehicle collision;

(UC04) Wrong-way driving: Integrates the trucks location data and the streets' direction within the port;

(UC05) Accidents with dangerous goods: Monitor dangerous goods being transported (according to the UN list of dangerous goods) in all use cases (1–4), adding adequate information regarding emergency procedures for effective response.

UC03 is particularly interesting because it requires the integration of data from both domains (health and logistics) and represents complex behaviors. For example, there is a possibility that bradycardia is detected followed by a continuous decrease in the heart rate after a collision. This situation reflects an accident where the driver is injured, classified as extremely severe with immediate urgency. In this situation, the vehicle collision is identified with both accelerometers from the ECG device and from the smartphone, considering device features as accuracy and energy consumption.

4.2 INTER-IoT-EWS: EWS Developed with SEMIoTICS

Our solution prototype (Fig. 3) includes the Shimmer ECG 3 device[7] to collect ECG data from drivers. This device has high accuracy and usability, being able to transmit data from a TinyOS application (running within the device) to a mobile phone application (Android) through Bluetooth. This mobile application receives and forwards the data to the cloud, acting as a gateway. Data are sent to the cloud and published in a broker as RDF/XML messages following the European Telecommunications Standards Institute (ETSI) Smart Appliances REFerence ontology (SAREF)[8] ontology extended with HL7 aECG (Annotated ECG), supported by the UniversAAL IoT platform.[9]

Similarly, the *MyDriving* mobile application for logistics (open use case of the Azure IoT platform[10]) transmits the data about the truck position, speed, accelerometer, and goods information to the cloud infrastructure. These logistics data are serialized as JSON messages, following the structure of SAREF ontology aligned to LogiCO ontology.[11] SAREF was chosen because of its capabilities for tracking devices' energy consumption. The IoT Platform Semantic Mediator (IPSM) module [22] is responsible for syntactically and semantically translating these data: from JSON and RDF/XML to the INTER-IoT JSON-LD syntax, which is structured JSON-LD (two @graph) with middleware information, and from SAREF to the INTER-IoT

[7] http://www.shimmersensing.com/products/ecg-development-kit.

[8] http://ontology.tno.nl/saref/.

[9] http://www.universaal.info/.

[10] https://azure.microsoft.com/en-us/campaigns/mydriving/.

[11] http://ontology.tno.nl/logico/.

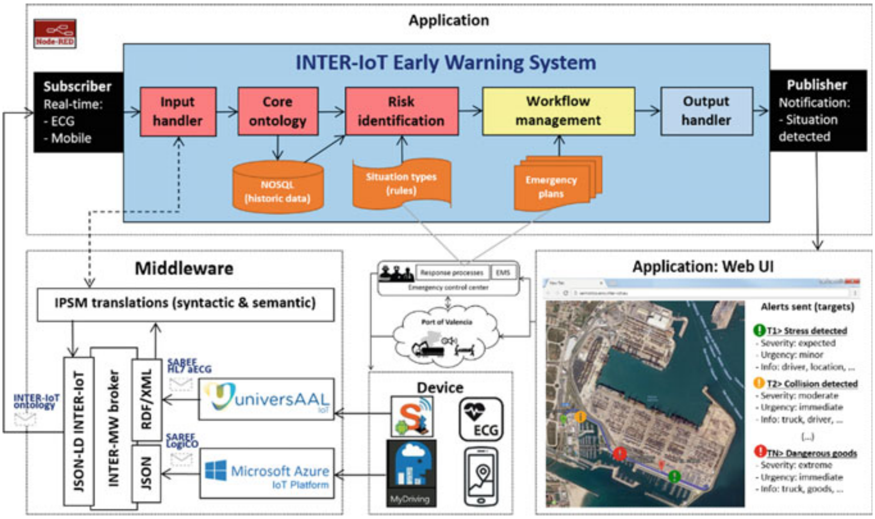

Fig. 3 EWS to detect accident risks and accidents at the port of Valencia

core ontology semantics, which is aligned with SSN. These translations are configured a priori in IPSM by the developer through a REST service.

The data represented as INTER-IoT JSON-LD syntax and INTER-IoT core ontology semantics are published in the broker in a topic, to which the EWS subscribes to receive real-time data. Then, the EWS input handler certifies whether new translations to harmonize the data in the SEMIoTICS core ontology are necessary and if so, the input handler requests the translations to IPSM.

Data are annotated with the core ontology and stored in a NoSQL database (MongoDB) as historical data. Both real-time data and historical data are used by the risk identification component, i.e., the SCENE CEP engine [20]. Situation types are defined a priori, as rule sets, describing the risky situations of interest based on emergency plans. Each situation type is linked to a response process, i.e., the specific workflow to be executed once a situation is identified. Therefore, the risk identification component triggers a workflow, which executes the linked processes.

The workflow component is responsible for checking the information requirements of each alert target, passing this information to the output handler, which is responsible for transforming the data to EDXL compliant semantically enriched messages. Therefore, the output handler enables the brokering of notifications of the situations detected, following the JSON-LD syntax and the EDXL data model, which is able to link to the used semantics. A web UI application shows each alert

Table 1 Data sources

External	Health	Logistics
Data	Driver's ECG, HR, accelerometer	Position, speed, accelerometer, goods
Device	Shimmer ECG 3 (capture), mobile	Mobile (MyDriving Android or iOS)
IoT platform	UniversAAL	MS Azure IoT
Ontologies	ETSI SAREF, HL7/aECG, FHIR	ETSI SAREF, LogiCO

sent by the EWS with its severity and urgency, and other information, including the targets that received the notification and the message sent to each target. The EWS is developed with Node.js and Node-RED. Table 1 summarizes the data sources used in the prototype.

4.3 Validation Plan

The validation plan of our solution follows the challenges *C1*, *C2*, and *C3* listed in Sect. 2.2 and is given in Table 2. It is organized as (a) factory acceptance tests (FAT): In a laboratory environment, the EWS is deployed in the cloud and the components integration tested through mock objects; and (b) site acceptance tests (SAT): a pilot in the port, where accidents will be simulated in accordance with the port emergency exercises. Both FAT and SAT assess whether the system works for the intended risks' detection and warning.

Since our approach is based on semantic translations, semantic loss at runtime will be used to calculate semantic interoperability, which will also be based on the semantic expressiveness of the EWS models (on specification level).

A comparison is included between our solution and a non-semantic approach for upstream data acquisition (from multiple devices), risks detection, and brokering. Thus, the plan includes the performance evaluation of data transfer, process, and brokerage. This plan includes data management according to the "Findable, Accessible, Interoperable and Reusable" (FAIR) data principles,[12] which enables research data to be reused.

Currently, the INTER-IoT-EWS is under implementation and testing phase. The initial execution and first results of A2 are presented in [19] and for A3 in [18]. Activities A1, A4, and A5 are ongoing. The first results of A4 show that using

[12] https://www.force11.org/group/fairgroup/fairprinciples.

Table 2 Validation activities

#	Activity	Description	Addresses
A1	Functional evaluation	Test cases with different levels of severity and urgency, checking the adherence with emergency procedures (pragmatic interoperability)	C1, C2, C3, FR1, FR2
A2	Semantic interoperability tests: semantic loss	Transformations in sequence from ontology A to ontology B and from B to A, i.e., check how x differs from $\mathrm{T}(\mathrm{T}(x)_{\mathrm{A>B}})_{\mathrm{B>A}}$, where $\mathrm{T}(x)_{\mathrm{A>B}}$ represents the semantic translation function from A to B [19]	C1, NFR1, NFR2
A3	Semantic interoperability tests: expressiveness	Specification level, i.e., how the models describe reality from different points of view [18]	C1, NFR1, NFR2
A4	Performance evaluation: data transfer	Compare JSON and JSON-LD as payload, measuring the impact of using JSON-LD rather than JSON, following the structure of the involved ontologies	C2, NFR3
A5	Performance evaluation: data process	Total time to translate; annotate data and insert into the database; access and process data for risk identification; and create alert messages (serialize as EDXL)	C2, NFR3
A6	Performance evaluation: data brokering	Scalability and resilience measured for single cluster and multi-broker, e.g., semantic IoT EWS approach [2], with sensor throughputs of up to 700 msg/sec	C2, C3, NFR3

JSON-LD brings an irrelevant burden to total transaction time when compared to JSON, and thus, migrating from JSON to JSON-LD is viable for the majority of the IoT solutions.

5 Conclusions

Current proposals for IoT-based EWS only partially address the semantic integration of a variety of data sources along with processing in time-critical applications and data analysis for effective responses. Our SEMIoTICS framework addresses this problem by applying different modeling languages, ontologies, and technologies toward the improvement of interoperability within and between IoT EWSs. To validate this framework, we are developing an EWS prototype, and we are currently applying it for detecting accidents at the port of Valencia.

Preliminary results include the INTER-IoT-EWS solution architecture, the required syntactic and semantic translations, and a validation plan guiding factory and site acceptance tests for measuring the interoperability support of the SEMIoTICS framework through the INTER-IoT-EWS. Initial tests indicate that the solution is adequate to cover the challenges, but this is an ongoing work to be reported in the near future.

The SEMIoTICS framework has been designed to be general enough to be applicable to other types of emergencies. However, the framework still lacks a mechanism to cope with the quality of information (QoI) at the network level, such as a Grubbs' test for outlier detection, and a statistical algorithm that can classify anomalous or invalid sensor values. Future work includes the development of a QoI mechanism and the completion of the execution of the validation plan.

Acknowledgements This work has been carried out under the CAPES funding BEX 1046/14-4 and EU-H2020-ICT grant INTER-IoT 687283.

References

1. UN. (2006). *Global survey of early warning systems: An assessment of capacities, gaps and opportunities toward building a comprehensive global early warning system for all natural hazards*. United Nations Report.
2. Poslad, S., et al. (2015). A semantic IoT early warning system for natural environment crisis management. *IEEE Transactions on Emerging Topics in Computing*.
3. Moreira, J. L. R., et al. (2015). Towards ontology-driven situation-aware disaster management. *Journal of Applied Ontology*.
4. Ganzha, M., et al. (2016). Semantic interoperability in the internet of things: an overview from the INTER-IoT perspective. *Journal of Network and Computer Applications*.
5. Comes, T., Mayag, B., & Negre, E. (2014). Decision support for disaster risk management: Integrating vulnerabilities into early-warning systems. In *Information systems for crisis response and management in mediterranean countries (ISCRAM-med)*.

6. Wächter, J., & Usländer, T. (2014). The role of information and communication technology in the development of early warning systems for geological disasters: The Tsunami show case. In *Early warning for geological disasters: Scientific methods and current practice*.
7. Alamdar, F., Kalantari, M., & Rajabifard, A. (2016). *Towards multi-agency sensor information integration for disaster management*. Environment and Urban Systems: Computers.
8. Raape, U., et al. (2010). Decision support for Tsunami early warning in Indonesia: The role of OGC Standards. In *Geographic information and cartography for risk and crisis management: Towards better solutions*.
9. Zambrano, A. M., et al. (2015). Sensor web enablement applied to an earthquake early warning system. In *International Conference on Internet and Distributed Computing Systems*.
10. Chronaki, C. E., et al. (2011). Interoperability in disaster medicine and emergency management. *Journal of Health Informatics*.
11. Ramar, K., & Mirnalinee, T. T. (2012). An ontological representation for Tsunami early warning system. In *IEEE-advances in engineering, science and management (ICAESM)*.
12. Barros, R., et al. (2015). EDXL-RESCUER ontology: An update based on faceted taxonomy approach. In *CEUR Workshop Proceedings*.
13. Middleton, S., et al. (2013). The seven main challenges of an early warning system architecture. In *Information systems for crisis response and management (ISCRAM)*.
14. Moreira, J. L. R., et al. (2016). Improving semantic interoperability of big data for epidemiological surveillance. In *I-ESA, BDI4E Workshop*.
15. Fertier, A., et al. (2016). Use of big data for continuous interoperability in crisis management. In *Enterprise Interoperability VII: I-ESA Proceedings*.
16. Mačinković, D., & Aničić, N. (2016). The systems development life cycle to facilitate progression towards semantic and organizational interoperability for healthcare system. In *Enterprise Interoperability VII: I-ESA Proceedings*.
17. Endsley, M. R. (1995). Toward a theory of situation awareness in dynamic systems. *Journal of Human Factors*.
18. Moreira, J. L. R., et al. (2017). Ontology-driven conceptual modeling for early warning systems: Redesigning the situation modeling language. In *MODELSWARD*.
19. Moreira, J. L. R., et al. (2017). Towards IoT platforms' integration: Semantic translations between W3C SSN and ETSI SAREF. In *Semantics. Workshop SIS-IoT*.
20. Costa, P. D., et al. (2016). Rule-based support for situation management. In *Fusion methodologies in crisis management: Higher level fusion and decision making*.
21. INTER-IoT. (2016). *INTER-IoT deliverable: D2.4*. Use cases manual.
22. Ganzha, M., et al. (2017). Streaming semantic translations. In *2017 21st international conference on system theory, control and computing (ICSTCC)*.