# Semantic Data Management: Sensor-based Port Security Use Case

Anna Fensel
and Michael Rogger
Semantic Technology Institute
University of Innsbruck
Innsbruck, Austria
Email: firstname.lastname@sti2.at

Tove Gustavi, Andreas Horndahl
and Christian Mårtenson
Swedish Defence Research Agency (FOI)
Stockholm, Sweden
Email: firstname.lastname@foi.se

***Abstract*—This work is part of the EU FP-7 project Support and describes a system-architecture to enhance port security by processing sensor data. The goal is to process vast amount of sensor readings of all types, reduce the noise in the data stream, cope with heterogeneities, detect patterns, fuse data streams and provide decision support in near real-time. We define an ontology to model the domain of sensors and events in the context of port security. The ontology is used as a common basis for our envisioned architecture. The architecture incorporates Sparkwave, a schema-enhanced pattern matcher, and Impactorium, a decision support system. Finally we present an evaluation approach for our use case and conclude with ongoing future work.**

***Keywords*—*port security, fusion services, semantic sensors, stream processing, stream reasoning, situation assessment, decision support.***

## I. Introduction

Ports are crucial for transportation and goods supply in Europe, providing a reliable and cost efficient infrastructure. Large ports such as the port of Gothenburg shipped in the year 2011 about 900k containers, 1.7 million passengers, 22.2 million metric tonnes of oil and 42 million metric tonnes of freight [1]. There are several ways the supply of goods can be interrupted, ports can be critical for the whole supply chain and can cause high economic loss. The most important problem ports face is theft, e.g., organized crime trying to gain access to containers. Also not to underestimate is the threat of terrorism, even though the probability is less for such events to happen, the potential loss is very high. Smuggling of goods such as explosives or chemicals is also one of many threats ports face.

The Support project, funded by the European Commission (EU FP7 Project 242112), and its partners aim to develop ICT-based support tools to increase the security of ports. The work presented in this paper focuses on increasing port security by integrating sensors and fusion services using semantic technologies. The paper describes a system architecture to process vast amount of sensor readings in near realtime. In the Support project we study the use of many different types of sensors, such as CCTV, IR cameras and different kinds of intrusion detection systems. The number of different information sources introduces heterogeneities which we deal with using an ontology. The ontology is designed in the context of port security and is based on RDF-S and on a fragment of OWL. Sensors produce most of the time noisy data due to the measurement of real world physical phenomena. The architecture should be able to reduce noise in the data stream. Filtering irrelevant data in an an early stage of the processing has the advantage that important data can be processed with more complex processing techniques in a timely fashion. Filtering data streams is performed by detecting patterns in the data stream. In order to help people in their decisions, matched patterns are combined with other non-sensor information to give the best possible decision support.

## II. Use Case

### A. Sensor-based seaside intrusion detection

The use case focuses on the problem of detecting an intruder from the sea-side of the port. In a simulated scenario an underwater passive tripwire is placed in the port of Gothenburg. The models are based on results from real trials [2]. The tripwire consists of sensors of two types of technologies, electromagnetic and acoustic, each with their strengths and weaknesses. The sensors are continously sending their readings to the stream processor, Sparkwave, which searches the stream for specific patterns. When a diver crosses the observed location, an intrusion pattern is activated and Sparkwave builds and transfers a report including location and concrete sensor readings.

### B. High-level fusion and decision support

As the underwater tripwire is very sensitive and the harbor is a busy environment, many false alarms are to be expected. To make sure that the report generated by Sparkwave is not a false alarm, it is fused with information from other sources. In the use-case we assume that the port authorities have received an intelligence report a week before saying that there is an increased risk of an attack against one of the ships in the port. The two pieces of information are automatically fused in the decision support tool Impactorium, resulting in the assessment that there is an increased risk of an ongoing intrusion from the sea-side.

## III. Related Work

The use of sensors, ontologies and semantic reasoning as a means for constructing general and flexible systems for

978-0-7695-5062-6/13 $26.00 © 2013 IEEE
DOI 10.1109/EISIC.2013.34

situation awareness has been recognized by many. In [3], Little and Rogova describe a general process for how to construct an ontology for situation and threat assessment for crisis management. Kokar et al. propose an ontology called Situation Theory Ontology (STO) for describing general situations [4]. The STO has been applied in the security domain where Fenza et al. demonstrate how it can be used to describe and reason specifically about airport security situations [5]. However, none of the above papers describe actual implementations, where the whole chain from sensor data to situation awareness is tested.

In [6] Tomic et al. describe a demonstrator system focussing on energy efficiency, based on ontology-based modeling and reasoning on top of sensor networks. In the security domain, Castro et al. report on an ontology-based system for intrusion detection using heterogeneous sensors [7]. However, their ontology is not based on standard semantic technologies, which in contrast to our work, makes it more difficult to maintain and integrate with other solutions.

## IV. Solution

### A. Architecture

The system presented in this paper has three main modules (Figure 1). The first is the sensing system, which in our case consists of two underwater sensor arrays, one electromagnetic and one acoustic. Each array generates a stream of sensor readings. Before the readings are sent to the next module they are converted to a semantic representation common to all modules. The second module is the pattern detection, consisting of the semantic stream processor Sparkwave. Sparkwave processes both input streams to detect combined patterns, e.g., when both streams independently indicate detection within a certain time frame. When a pattern is detected, Sparkwave generates a new message in the common semantic representation format and feeds it to the risk assessment module. This module consists of Impactorium, a decision support tool where a user can define, model and assess situations or threat events. When a new message reaches Impactorium, it is automatically queried for *indicators*, information that influences the assessments of the modeled situations. If there is an indicator match, the corresponding model is updated according to the new information leading to an updated situation assessment and a possible alert to the user.

For performance reasons, the components of the solution are integrated using loosely coupled RESTful web services. In a real world scenario where numerous sensors are involved, standard load-balance techniques can then be applied. For instance, Sparkwave and the calculation services used by Impactorium could be installed on multiple machines and process information in parallel. Each service in this setup would handle a unique subset of detection patterns.

### B. Ontolgy

For the specific needs of the use case, a small tailor-made event ontology has been developed. The ontology was constructed using RDF-S and a limited set of features from OWL (transitive and inverse object properties). The ontology consists of a class hierarchy (e.g., Diver is a sub-class of Actor which is a sub-class of Object), a number of class attributes or data properties (e.g., the class Event can have the attribute ”event-start-time”), and a number of relations or object properties (e.g., the service that generated a specific report can be linked to that report through the relation ”report-generated-by-service”).

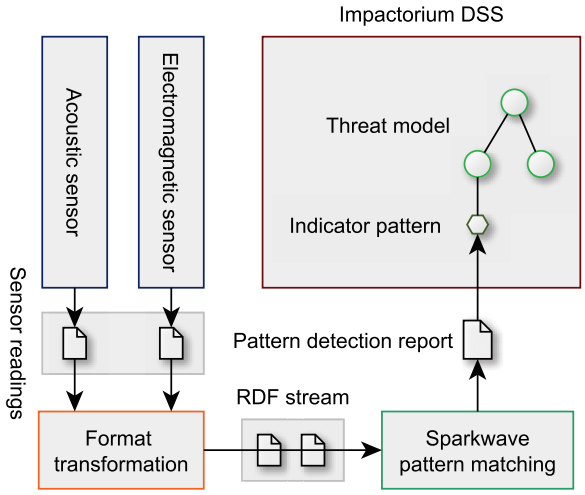


Fig. 1: Overview of the architecture

### C. Sparkwave

Sparkwave is a scalable solution to perform continuous schema-enhanced pattern matching over RDF data streams. More precisely, the goal of Sparkwave is to provide efficient pattern matching functionalities on RDF streams in a truly continuous way, enabling the expression of temporal constraints in the form of time windows and taking into account RDF schema entailments [8]. As a component, Sparkwave is used to filter high-frequency noisy data and heterogeneous observations produced by sensors. Experiments show already high throughput on commodity hardware [8], which makes it predestined to process a high volume of sensor readings in near real time. The incorporation of heterogeneity e.g., the sensor type, sensor properties and locations are tackled using integrated semantic technologies: RDF and lightweight reasoning. Existing sensor readings are enhanced by inferring knowledge using pre-defined ontology expressed in RDF-S and OWL property “inverseOf”. Additional to entailments also static data is used to enhance the data stream. Static data typically does not change over a longer period of time. For example the exact location of the sensors is static because it is assumed to not change, therefore can be kept in memory over longer periods of time. Also possible is to structure location data in a hierarchical way, e.g., “SensorA” and “SensorB” are both part of “FacilityX” which itself is part of “AreaY”. This enables to define more flexible and abstracted queries in the pattern, for example queries which match all sensors located in “AreaY” and correspond to certain criteria.

### D. Impactorium

Impactorium is a model-based decision support tool that is used to compute estimated threat levels for a number of

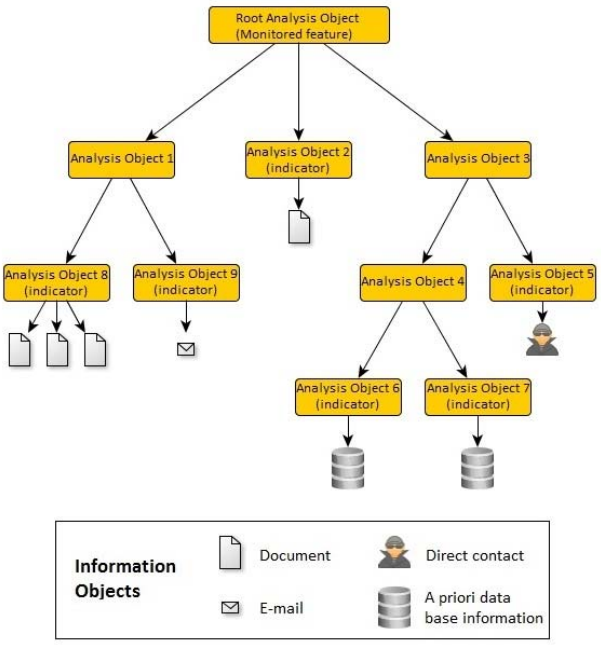


Fig. 2: Example showing the tree structure of an Analysis Model in Impactorium. The top node is associated with a threat hypothesis, which is analyzed and broken down into components.

pre-defined threats [9], [10]. The idea behind Impactorium is to make use of the fact that a threat is not an isolated event. Even if no first-hand information about the threat level is available, it may be possible to assess the threat level by combining information about related events and features. With Impactorium it is possible to make this sort of high-level information fusion automatically in real-time. This is especially practical for threats that have to be constantly monitored.

For each threat that is to be monitored, Impactorium needs a model that specifies, in terms of logical statements or mathematical formulas, the dependencies between threat level and the different types of information that are possible to obtain. The models in Impactorium are in the form of tree structures. An example is shown in Figure 2. When a potential threat has been identified, a threat hypothesis is formulated. A threat hypothesis could for instance be ”criminal group X is executing a container theft” or ”organization Y is planning a sabotage against a shipping of goods to company Z”. The threat hypothesis defines the root node in a new tree structure. A subject matter expert then analyzes the threat in order to find related events or conditions which are formulated as true/false statements. These make up the second layer of nodes in the tree structure. A statement such as ”X have access to restricted information about a shipping” could generate a set of new nodes in the tree model, representing different ways X could have gained access to that information. In the end, the model can consist of many ”branches” and several layers of nodes. The purpose of the segmentation is to break down every branch in the tree structure to the level where the true/false statements associated with the bottom nodes can be answered, based on information that is obtainable from available information sources (sensors, intelligence reports, etc.). The states of the bottom nodes are propagated up through the fusion functions in the model and are indirectly used to compute the belief value of the threat node.

## V. Evaluation Approach

In the following we describe an evaluation approach. A data set, consisting of a dynamic and a static part is extracted for our use case detailed in Section II. Additionally we describe the Sparkwave stream pattern and the indicator rules for Impactorium.

### A. Data Set

*a) Dynamic:* The sensor readings used cover the time span from August 30th, 2011 10:00:05,097 till August 30th, 2011, 10:21:25,943. There are in total 53,368 readings coming from acoustic sensors and 8,168 readings from electromagnetic sensors. The readings are bundled in batches of 8 readings per batch where all readings in a batch are sharing the same time stamp.

*b) Static:* A part of the raw sensor readings data conveys the concrete location about sensor position. Since this information is static we choose to extract it from the stream and store it in a separate file. During a Sparkwave instance boot up such static instances are processed and kept indefinitely in the networking structures during stream pattern matching.

### B. Stream Pattern

The Sparkwave pattern used in the use case searches for an occurrence of signals coming from two distinctive sensor types at the same location. After detecting an occurrence the pattern is packaging related pattern instance data into a report and sent to Impactorium over a REST-based interface. The pattern is shown in Listing 1. Note that prefixes, handlers, construct and literal typing are omitted for readability.

```
WHERE {
?detection1 wp4:has_status ”true” .
?detection1 wp4:has_sensor ?sensor1 .
?sensor1 rdf:type wp4:PETSensor .
?sensor1 wp4:sensor_has_location ?loc1 .
?loc1 wp4:location_is_part_of_location wp4:DockX .
?detection2 wp4:has_status ”true” .
?detection2 wp4:has_sensor ?sensor2 .
?sensor2 rdf:type wp4:PATSensor .
?sensor2 wp4:sensor_has_location ?loc2 .
?loc2 wp4:location_is_part_of_location wp4:DockX .
?detection1 wp4:event_reported_in_report ?sr1.
?detection2 wp4:event_reported_in_report ?sr2 .
TIMEWINDOW (100) }
```

Listing 1: A Sparkwave pattern to detect electromagnetic and acoustic sensor in the same location and time-window

### C. Indicator rules

When Impactorium receives a report from Sparkwave the message is examined to determine if it can be associated to any risk model indicators. In the use case, we only have one model defining how to fuse manual observations, sensor observations

and intelligence information to assess the risk of a seaside intrusion (Figure 3(a)).

The intelligence node defines a context value which represents the current security level. The values of the intelligence node and the manual observation node are set manually. However, what is of interest for this paper is the sensor detection node, which is set automatically based on rules. Each rule defines 1) a pattern that determines if the incoming report is relevant to the indicator 2) a value that the indicator will take if the report matches the pattern. In our case, the rules are defined as follows. If Impactorium receives a report based on simultaneous (or close) detections from both sensor types, the Sparkwave report is tagged with "probability high" and the indicator value is set to a high value, in our case 1. If only one of the sensor types is behind the detection, another Sparkwave pattern is triggered and generates a message tagged with "probability low". In this case the indicator value is set to a lower value, in our case to 0.7. An example of an indicator pattern is shown in Listing 2.

```
ASK {
 ?detection rdf:type support:DiverIntrusion.
 ?detection support:detectionProbability "high"
 ?detection support:location ?location.
 ?location support:location_is_part_of_location
     support:DockXWater .
}
```

Listing 2: Example of an indicator pattern

Figure 3(b) shows the assessment before detections are made. The resulting value of 0.3 is the mean value of the intelligence node and the observation node. When the message from Sparkwave triggers the rule the sensor detection node is set to 1, which propagates via a max-rule for the observation node (it is enough that either a sensor or a manual detection is made) to give a new value of 0.8 for seaside intrusion (figure 3(c)).

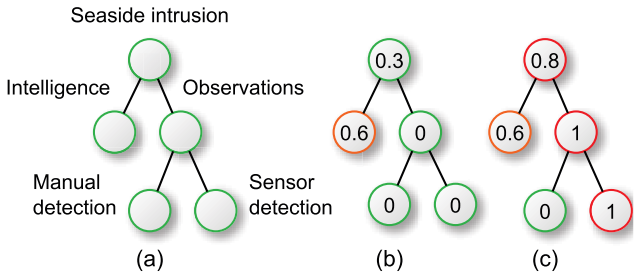


Fig. 3: A fusion model for seaside intrusion (a). The leaf nodes are the observable indicators whose values propagate upwards through simple mean, max and min functions when new information arrives (b and c).

## VI. Future Work

This paper describes the initial steps of implementing an end-to-end solution for semantic integration of sensors and fusion services in a port security setting. The next steps will be to gradually expand the experimental setting with additional sensors and signal processing capabilities and include more stream processing patterns and threat models. The ontology will be updated to cover the new settings. We also plan to evaluate the system as a whole, both offline using the previously mentioned data-sets and online using a live system with data streams from sensors placed at the port.

## VII. Conclusion

In this paper we presented an approach to enhance security of ports by processing vast amount of sensor data provided as streams. We defined a unifying ontology to model sensors and events in the context of port security, the model is considered as basis for the system architecture. The envisioned architecture consists of Sparkwave and Impactorium. The architecture aims to reduce noise in the data stream, cope with heterogeneities, detect patterns, fuse data streams and provide decision support in near real-time. We sketched an evaluation approach for our architecture. In the future, we plan to extend gradually the experimental setting with additional sensors, patterns and threat models.

## Acknowledgment

This work has been supported by the European Commission through the SUPPORT project (EU FP7 Project 242112) and by the R&D program of the Swedish Armed Forces. The authors wish also to acknowledge Srdjan Komazec for his effort spent on Sparkwave and the Support project.

## References

[1] "Port of gothenburg in short," 03 2013. [Online]. Available: http://www.portofgothenburg.com/About-the-port/Fact-file-Port-of-Gothenburg/

[2] R. Lennartsson, E. Dalberg, and S. Petrovic, "Underwater intruder detection with passive tripwires," in *Waterside Security Conference (WSS), 2012 International*. IEEE, 2012.

[3] E. G. Little and G. L. Rogova, "Designing ontologies for higher level fusion," *Information Fusion*, vol. 10, no. 1, pp. 70–82, 2009.

[4] M. M. Kokar, C. J. Matheus, and K. Baclawski, "Ontology-based situation awareness," *Information fusion*, vol. 10, no. 1, pp. 83–98, 2009.

[5] G. Fenza, D. Furno, V. Loia, and M. Veniero, "Agent-based cognitive approach to airport security situation awareness," in *Complex, Intelligent and Software Intensive Systems (CISIS), 2010 International Conference on*. IEEE, 2010, pp. 1057–1062.

[6] S. Tomic, A. Fensel, and T. Pellegrini, "Sesame demonstrator: ontologies, services and policies for energy efficiency," in *Proceedings of the 6th International Conference on Semantic Systems*. ACM, 2010, p. 24.

[7] J. Castro, M. Delgado, J. Medina, and M. Ruiz-Lozano, "Intelligent surveillance system with integration of heterogeneous information for intrusion detection," *Expert Systems with Applications*, vol. 38, no. 9, pp. 11 182–11 192, 2011.

[8] S. Komazec, D. Cerri, and D. Fensel, "Sparkwave: continuous schema-enhanced pattern matching over rdf data streams," in *Proceedings of the 6th ACM International Conference on Distributed Event-Based Systems*. ACM, 2012, pp. 58–68.

[9] R. Forsgren, L. Kaati, C. Mårtenson, P. Svenson, and E. Tjörnhammar, "An overview of the impactorium tools 2008," in *Proceedings of the Second Skövde Workshop on Information Fusion Topics (SWIFT 2008)*. Citeseer, 2008.

[10] P. Svenson, R. Forsgren, B. Kylesten, P. Berggren, W. R. Fah, M. S. Choo, and J. K. Y. Hann, "Swedish-singapore studies of bayesian modelling techniques for tactical intelligence analysis," in *Information Fusion (FUSION), 2010 13th Conference on*. IEEE, 2010, pp. 1–8.