A SMART MONITORING SYSTEM BASED ON A FAST CLASSIFIER AND A SEMANTIC POST REASONER

Flora Amato, Valentina Casola, Mariana Esposito, Nicola Mazzocca, Antonino Mazzeo

Dipartimento di Informatica e Sistemistica
Università di Napoli Federico II

{flora.amato,casolav,mariana.esposito,nicola.mazzocca,mazzeo}@unina.it

ABSTRACT

In modern decision support systems there is the need to improve the performance in terms of detection, reliability and real time capabilities. These features are usually in inverse proportion. In this paper we propose an innovative approach for a smart event detection and enriched phenomena comprehension. In particular, the proposed approach is based on a two steps process that tries to quickly identify an alarm and then elaborate the acquired knowledge base with a post reasoner to refine the final decision and give operators more feelings about the situation assessment and raised alarms.

Keywords: monitoring systems, semantic based, knowledge base, sensor data models.

1. INTRODUCTION

The pressing need for territorial protection and for the deployment of suitable disaster prevention strategies has led the protection agencies, as well as the international scientific community, in an effort aimed at the definition of new homeland security strategies and tools. A central activity in any Homeland Security system is the monitoring and observation of different phenomena, aimed at providing an updated and meaningful description of the monitored scenario, as well as its possible evolutions, to enable proper countermeasures for the protection and safety of people and things. In these scenarios, not only smart surveillance and alert systems are needed but enriched decision support systems (DSS) are desirable. Such systems rely on heterogeneous data acquisition tools (sensors, video, historical and simulated data, …) and on data elaboration to prune non significant information; nevertheless, this is not enough as there is the need to interpret what data really represents to reduce false positives and detect even weak alarm conditions. The availability of advanced monitoring techniques and heterogeneous information sources has increased the accuracy in observing, measuring and describing the nature of phenomena: the current level of technology in this field represents an opportunity to improve the understanding about observed phenomena but, at the same time, it introduces a high degree of complexity in the data elaboration and fusion.

Intelligent decision support systems are necessary to enable, when possible, the automatic adoption of countermeasures in case of alarms or to support end users during decision making activities (when a too large number of sensors, devices, or cameras placed inside the site to be protected produce a wide amount of data to be processed). However, many automatic and intelligent detection systems generate unnecessary warnings (false alarms); this problem, unfortunately, severely limits the use of these systems to enable automatic or partially automatic counter-measures. In recent years, scientific world's attention has been devoted to both the information management with information and decision fusion approaches, and to the quantitative reliability estimation of these systems.

On the other hand, to improve the situation assessment, it is possible to adopt different types of models for description of knowledge-base, event correlation and for the definition of the situation and threat identification. Very promising approaches are based on semantic and ontological models.

The semantic model can be used for understanding observed phenomena. In particular all sensors must share the same data model and the same interpretation of data. The data model must provide a syntactic interoperability mechanisms and procedures for semantic enrichment to build models in order to (i) ensure a correct and shared information interpretation, (ii) aggregate raw data into events (simple and composed), that will be used for the situation assessment before a final decision.

In the literature some approaches for event detection and decision support based on semantic inference rules for phenomena comprehension are available. Nevertheless, due to the introduced overhead, the knowledge base is just inferred in offline mode.

In this paper we propose an innovative approach for smart event detection and enriched phenomena comprehension: the knowledge base will be inferred in real time, for the event detection, and a light smart classifier will raise an alarm. The proposed approach is based on two steps:

1. A smart and light on-line inference engine to raise an alarm, in case of threat event detection;
2. A post reasoner off-line inference engine, in order to comprehend the event and its causes.

The former has the task to detect, with real time constraints, dangerous condition, giving a pre-alarm; the latter performs a more complex reasoning activity in order to help users to comprehend the dangerous situation and refine the decision.

The reminder of the paper is structured as follows, in Section 2 some related works are reported, in Sections 3 and 4 a model and relative architecture of the proposed monitoring system are presented. In Section 5 a simple case study on the smart classifier is presented and, finally, in section 6 some conclusions are discussed.

2. RELATED WORKS

In the literature some semantic approaches to manage heterogeneous data from sensors and to infer them for event detection are available. These approaches exploit offline inference in order to extract implicit knowledge from data sensor.

On the other hand, some approaches are beginning to use in line techniques both for enrich the semantic data model and to manage in real time event detection; very often, an offline inference for event comprehension and phenomena analysis is associated.

In (Huang and Javed, 2008) an architecture for sensor information description and processing, named SWASN (Semantic Web Architecture for Sensor Network), is proposed. The architecture is based on four layers: the first is the physical level composed by different sensor networks. Each sensor networks manage its own data format. The data are processed in an Ontology Layer, in which each network has a local Ontology. A Global Ontology is built upon a common vocabulary and it is processed in the Semantic layer for the knowledge extraction, through inference and semantic reasoning. Finally, at user level, it is possible to query the ontology in order to process and elaborate data.

Similar architectures are presented in (Gomez and Laube 2009; Konstantinou, Solidakis, Zoi, Zafeiropoulos, Stathopoulos and Mitrou 2007; Konstantinou, Solidakis, Zoi, Zafeiropoulos, Stathopoulos and Mitrou 2010). In particular in (Gomez, and Laube 2009) and (Solidakis, Zoi, Zafeiropoulos, Stathopoulos and mitrou 2007) an automatic process for transformation of XML data into RDF is proposed, the transformation process is driven by semantic reasoning and mapping rules. The transformation is in real-time but not any detection system is proposed. In (Konstantinou, Solidakis, Zoi, Zafeiropoulos, Stathopoulos and Mitrou 2010) a middleware architecture to manage event detection in real time is presented. It is a middleware architecture for automated, real-time, unsupervised annotation of low-level context features and corresponding mapping to high-level semantics. It enables the composition of simple rules through specific interfaces, which may launch a context aware system that will annotate content without the need for user technical expertise. The middleware has a semantic model only for the event management. There are no models for the data acquired by sensor.

3. A MODEL FOR MONITORING SYSTEM

The proposed approach combine significative results available in literature to enrich data models with semantic information and, at the same time, use a smart classifier to let the detection process be quicker.

A monitoring system can be composed of two main layers: the sensor network and the monitoring system (Casola, De Benedictis, Mazzeo and Mazzocca 2011). As illustrated in figure 1, the sensors network can be, in turn, characterized by: *Sensors Physical Features*, *Measurement Typology* and *Topology*.

The monitoring system, based on inference engines, can be characterized by: *Real Time Acquired Knowledge*, *Real Time Inferred Knowledge* and *Post Reasoner Knowledge*. Each sensor node is responsible to measure specific parameters. The *Sensor physical features* layer models the physical characteristic of a single sensor node and of the whole sensor network.

Measurement Typology layer defines what kinds of measures are gathered from the sensors.

Topology layer models information about the system deployment, describing how the sensors are located in the area of interest.

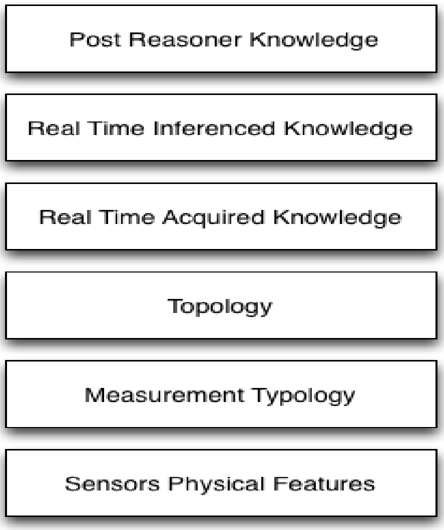
Post Reasoner Knowledge
Real Time Inferenced Knowledge
Real Time Acquired Knowledge
Topology
Measurement Typology
Sensors Physical Features

Figure 1: Model layers for a Monitoring System

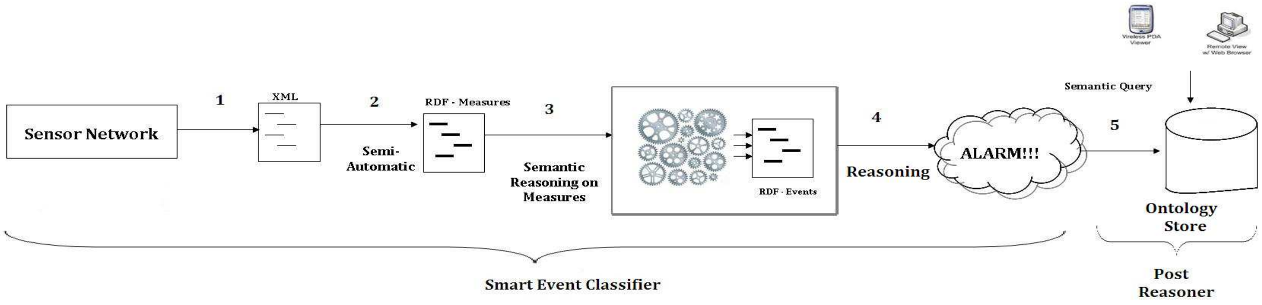

Figure 2: The reference Architecture

The monitoring system acts at two different levels: a real-time reaction and an off-line (post reasoner) activity. The first aiming at providing proper alarms when dangerous events occur, the second aiming at providing a complete and detailed picture of the situation that can be useful for operators both in understanding the situation and for decision supporting.

So, the monitoring system works in these two steps, by means of a fast classifier and through a post reasoner. It can be modelled, in turn, by three layers.

The *Real Time Acquired Knowledge* layer has the task of modeling the typology, the structure and the values of raw and structured data acquired and transmitted by sensors, enriched with semantic information about them (Amato, Casola, Gaglione and Mazzeo 2011). At this level the data is modelled and processed by both the fast classifier and the post reasoner.

The encoding language, used to process and transmit information is the RDF standard.

The *Real Time inferred Knowledge* layer models the knowledge derived by the application, on the sensed data. This level already works on semantically enriched data, the overhead for linking data with information about them, in fact, is necessary at this level because there is a multitude of events that can be detected only by combining information from collections of sensors that are heterogeneous both for typology of measurement carried out and for the data format in which they are sent to centralizer nodes.

Furthermore, at this level, many details on the current situation are abstracted away, in order to allow the classifier to perform efficient decision tasks, even if it is not able to derive the full knowledge about the monitored environment. The cut information is then re-considered into the abstract model of the *Post Reasoner Knowledge*, which works without real time constraints. The *Post Reasoner Knowledge* layer aims at modelling all relevant aspects of the monitored environment. It is focused to derive useful knowledge to have a detailed view of the situation, finalized to :

- help in situation awareness
- support in the decision process.

The acquired data are enriched with RDF semantic information and processed by a reasoner based on Pellet (Kaplanski 2012).

The reasoner is based on a general rules component and a specialist component implementing the rules tuned on the environment to be monitored.

The relevant domain knowledge is encoded with the help of domain experts using appropriate data structures, the ontologies which model the elements of interest in terms of concepts and relationships relating to the phenomena to be monitored, the events and the associated actions to be performed. This ontology is used in order to link each element outputted by the reasoner with a proper descriptions and appropriate information that can be exploited for helping users to understand the situation. The system implementing the semantic reasoner is much more computational expensive than the classifiers used for the real time decisors. At this level, in fact, the outputted inferred data is designed to give support to users with offline reasoning and data mining features, which can be exploited to get a complete knowledge of the situations, even at a later time.

4. REFERENCE ARCHITECTURE

The architecture proposed to implement the proposed monitoring system model, combines different approaches available in the literature. As illustrated in Figure 2, it is composed of:

- A Smart Event Classifier (implementing the Real Time Acquired and Inferred Knowledge layers);
- A Post Reasoner (implementing the Post Reasoner Knowledge layer).

We implemented a fast classifier in order to detect, with real time constraint, potential dangerous condition and then, if necessary, raise an alarm. The events detection is carried out by data correlation coming from different sensors. As a matter of fact, in real situations the potential hazard cannot be detected by using data coming from a single device.

The classifier has a standard structure, composed of learner and predictor components, to build a predictive model and exploit this model for event detection. The predictor is responsible to classify the data collection coming from the sensor network in order to decide if alarm conditions have occurred.

We adopted a rule-based classifier implemented as a decision tree. As usual, in decision tree mechanism, the set of decision rules is modelled as a tree in which leaves represent class associated to the events to be detected and branches represent conjunctions of features, i.e. condition on the sensed data, that lead to those event classes.

In order to define the branch rules domain experts manually classify a sectioned set (training set) of event data. These data are used by the learner module in order to set the predictor parameters, which regulate the automatic detection of the alert conditions. To increment the system performance, the rules have been pruned recurring to a manual refinement made by domain experts (Liu, Ma and Young 2000).

In figure 3 we report a small example of rule codified as a tree branch. The codified rule is:

```
(S1.location = 41°53’24” N, 12° 29’ 32” E,
S1.Pressure > 101.325 kPa,
S2.Pressure > 30 inHg,
S2.location= 41° 53’ 37”N,12°29’11” E)
==>Alarm
```

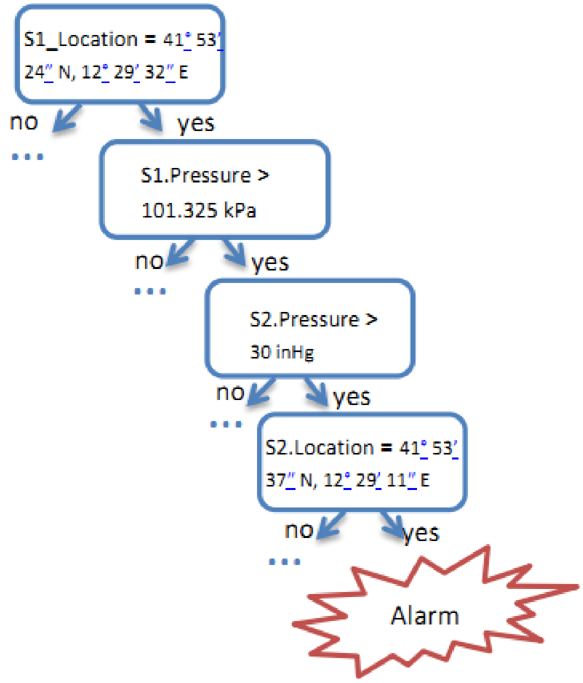
S1_Location = 41° 53’ 24” N, 12° 29’ 32” E
no
yes
...
S1.Pressure > 101.325 kPa
no
yes
...
S2.Pressure > 30 inHg
no
yes
...
S2.Location = 41° 53’ 37” N, 12° 29’ 11” E
no
yes
...
Alarm

Figure 3: Tree branch codified Rule

Predictor is realized as a parametric system, whose parameters may vary, or in the presence of new data that affect the system output, or by the intervention of the operator that can decide to manually change them for the occurrence of new conditions.

To fulfil the real time constraints, allowing the system to react in useful time, the classifier may be synthesized in hardware; in particular, in order to manage the variations on the parameters, the predictor is synthesized on a reconfigurable device (FPGA) that allows reconfiguration of the system when is necessary (Wittig and Chow 1996). From a semantic enrichment point of view, the smart Event classifier performs the following actions:

1. Sensor networks gather data and format them in XML files (Konstantinou, Solidakis, Zoi, Zafeiropoulos, Stathopoulos and Mitrou 2010), they encode both sensor properties and measured values;
2. With a semi-automatic activity, XML files are semantically enriched and transformed in RDF files. This file is compliant with the domain ontology and it is suitable to perform semantic reasoning;
3. With the SWRL (Horrocks, Patel-Schneider, Boley, Tabet, Grosof and Dean 2004) language it is possible to perform different inferences to compose simple events in composed ones.
4. On composed events it is possible to detect threats events and generate alarms.

These actions are performed in real time; they also feed the knowledge base for the Post Reasoner component. The just built ontology is stored in a repository (Triple Stores) (Broekstra, Kampman Van Harmelen 2002) and can be used off-line through the adoption of semantic query languages as SPARQL (Prud’hommeaux and Seaborne 2004). Through queries, the post reasoner is able to understand and explain to end users the meaning of the alarms and their causes.

The Knowledge Base can be seen as an information repository about a particular domain of interest. Typical knowledge bases consist of concepts, properties and instances. We encoded the knowledge base using the ontology. The ontology is a set of Classes, Properties and Instances. The Classes define the domain concepts; the Properties define the relation between Classes (Domain to Range). The properties can be between two classes or attributes (a property of a class).

During reasoning, inferences are made, classifying instances of the ontology and associating new properties to instances while maintaining logical consistency.

The reasoner, based on Pellet (Kaplanski 2010), is able to infer logical consequences from a set of asserted facts about the monitoring system defined by user experts. In particular it is composed of two components, one implementing the general inference rules and one the specialist rules, defined by domain experts in order to capture the relevant knowledge about the environment to be monitored.

The system uses first-order predicate logic to perform reasoning. The inferences proceed both by forward chaining and backward chaining (Kaplanski 2010). Not having real time constraints, the post reasoner is not necessary implemented on an embedded system.

5. CASE STUDY

In this section we provide an application of our system for the monitoring of a subway station. The station is supervised through different sensor technologies (Smart-cameras, Infrared Sensors, etc…). The correlation of the different measures, gathered by the sensors, allows to detect some events (e.g. physical intrusions, explosions,…) and, if necessary, raise a proper alarm to the operator.

The station is equipped with a security system including intelligent cameras (*S1*), active infrared barriers (*S2*) and explosive sniffers CBRNe (Chemical Biological Radiological and Nuclear Explosive) (*S3*) for tunnel portal protection. The attack scenario consists of a sequence of simple events which should be detected by the appropriate sensors and combined in order to form the composite event.

The actors of the scenario are defined through instances and they belong to the classes. We implemented an ontology represented in figure 4. This ontology aims at representing the domain of interest, including measures, events and alarms.

Sensor class has a subclass for each device sensor, in this specific case we use an Infrared barrier sensor (IR), a chemical explosive detection sensor (CBRNe) and an intelligent camera (IC) in which are implemented algorithm for video content analysis. The Detect_Event class represents events detected by correlating measurements from the different sensors. Events can be simple or composed. Simple events are related to events detected by single sensors, such as presence of Train detected by smart camera. Composed events are a combination of simple events by means of proper rules. Some composed events can generate an alarm in case of activation. Furthermore, a sensor measures some parameters in order to detect an event; it is characterized by a location and typology. The data properties for some classes are described in table 1.

Table 1: Class and Data Property.

Class	**Data Property: type**
Sensor	ID: int
Measurement	AtTime: DateTimeStamp
Detect_Event	DetectTime: DateTimeStamp

Ontology instances are constantly updated and populated through reasoning operations. The rules allowing the population and enrichment of the ontology are of the following typology:

1. Reasoning on the measures for detect events;
2. Reasoning about simple events in order to generate compounds events;
3. Reasoning on the events for alarms generation.

In this example we show the detection of the "Drop_Explosive_Tunnel" event, regarding the release of explosives in an underground tunnel. In the case of event trigger, a proper alarm must be raised.

Let us suppose that the dynamic of the scenario follows the steps reported below:

1. The attacker stays on the platform for the time needed to prepare the attack, missing one or more trains;
2. The attacker goes down the tracks by crossing the limit of the platform and moves inside the tunnel portal;
3. The attacker drops the bag containing the explosive device inside the tunnel and leaves the station.

A specification for these events is in the following:

- E1. extended presence on the platform (E1 by *S1*);
- E2. train passing (E2 by *S1*);
- E3. platform line crossing (E3 by *S1*);
- E4. tunnel intrusion (E4 by *S2*);
- E5. explosive detection (E5 by *S3*).

The combined event "Drop_Explosive_Tunnel" can be specified in two ways as follow:

1. If (E1, E2) then (E4, E5)
2. If E3 then (E4, E5)

Where E1, E2, E3 and E4 are simple events. The clause "then" states a temporal sequence for the event detection. For brevity, we show the first node activation. The sensed data are firstly codified in XML format (Listing 1 in Appendix), by the centralized nodes implemented in the *Real Time Acquired Knowledge* layer. The listing contains basic information about a sensor, ID, performed measurements, temporal information and value data. In particular, the sensor CBRNE, detecting the presence of an explosive (value = true), is reported. This information is then semantically enriched by exploiting the proper domain ontologies. Starting from the XML, a RDF file is then produced (Listing 2 in Appendix). In the listing the sensor CBRN1, instance of CBRNE class, is reported, it processes information of Chemical type and is positioned in the station 1 (S1). In the same listing CHEM2, instance of the Chemical Class (sub-class of Measurement) is reported. Moreover, the instant of the measurements (AtTime Date) and the value (hasChem) are reported. In this case, the conditions allow the smart classifier to infer the presence of explosive event, in fact, it firstly detects simple events, compose them and raise the alarm condition. The composition of simple events produces the following compounds events:

1. (E1, E2)-> Dangerous_Presence
2. (E4, E5) -> Possible_Explosive

Condition 1 states means, if both events E1 and E2 occur in the same time, then "Dangerous_presence" event is triggered, the second one states if E4 and E5 events occur, the composite event "Possible_Explosive" is detected. The combination, with temporal constraints, of "Dangerous_Presence" and "Possible_Explosive" events triggers the "Drop_Explosive_Tunnel" event, launching the corresponding alarm. In Listing 3 is reported the activation event E5 "Detect_Explosive", triggered by condition on "is_Explosive_Detection". In the second

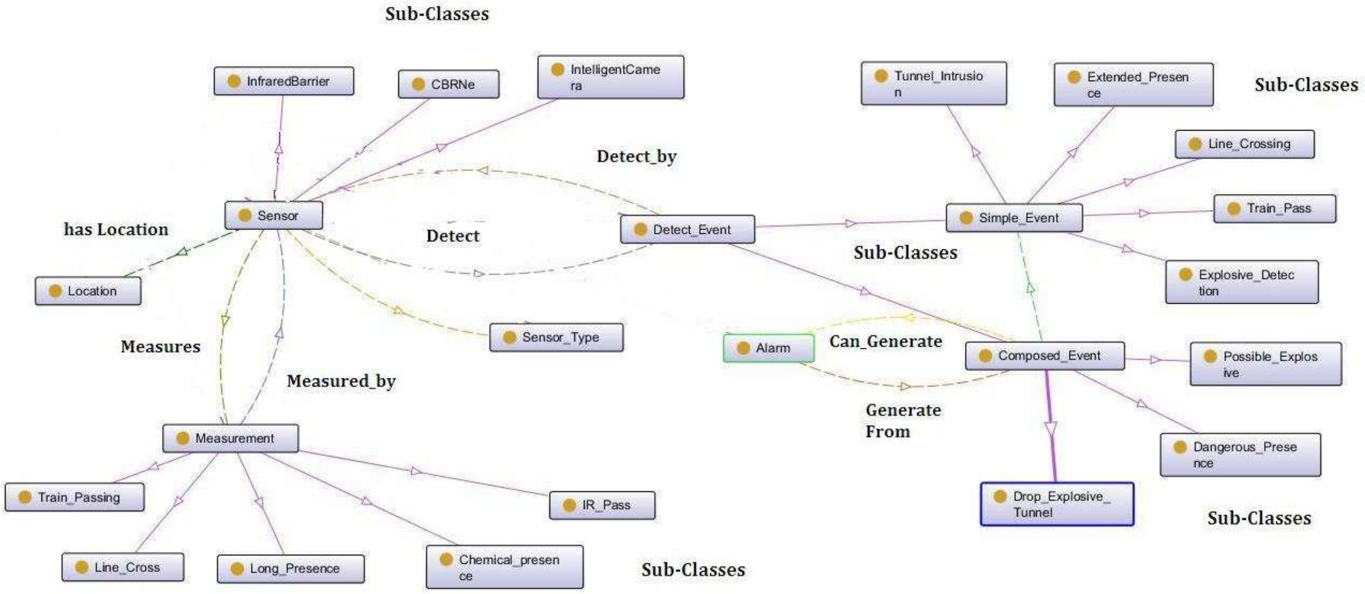

Figure 4: The ontology

part of the listing the event "Drop_Explosive_Tunnel" is composed as composition of "Dangerous_Presence" and "Possible_Explosive" that occur in temporal succession. Finally, Listing 4 shows the activation of the alarm caused by the "Detect_Drop_Explosive" event. The conditions used to manage and understand the cause of the alarms may be queried off-line, through a user friendly interface that exploits SPARQL language for querying the semantic enriched data about the situation, as the alarms that have been triggered and the events detected.

6. CONCLUSIONS AND FUTURE WORK

In this paper we proposed a smart monitoring system based on a two steps process that tries to quickly identify an alarm and then elaborate the acquired knowledge base with a post reasoner to refine the final decision and give operators more feelings about the situation assessment and raised alarms. The proposed approach combine significative results available in literature to enrich data models with semantic information and, at the same time, use a smart classifier to let the detection process be quicker. We have illustrated a simple example, primary focused on the enrichment process to build the knowledge base on which a post reasoner can infer further information for situation assessment. In future works we intend to complete the architecture implementation with this component, too, and use this approach to enrich available decision support systems that are based on different detection models as statistical or mathematical ones.

APPENDIX A - LISTING

```
<?xml version='1.0' encoding='UTF-8'?>
   <result>
    <nodeid value='1'/>
    <location value='Station1'>
    <name value='Chemical_Presence'/>
    <data value='true'/>
<timestampvalue='2012-05-13T09:00:03+01:00'/>
    </result>
```

Listing 1: example of XML sensor output

```
<!-- http://www.owl-
ontologies.com/Ontology1.owl#CBRN1 -->

 <owl:NamedIndividual
rdf:about="http://www.owl-
ontologies.com/Ontology1.owl#CBRN1">
   <rdf:type
rdf:resource="http://www.owl-
ontologies.com/Ontology1.owl#CBRNe"/>
   <hasType rdf:resource="http://www.owl-
ontologies.com/Ontology1.owl#Chemical"/>
   <hasLocation
rdf:resource="http://www.owl-
ontologies.com/Ontology1.owl#Station1"/>
 </owl:NamedIndividual>

<!--http://www.owl
ontologies.com/Ontology1.owl#Chemical -->

<owl:NamedIndividual
rdf:about="http://www.owl-
ontologies.com/Ontology1.owl#Chemical">
  <rdf:type rdf:resource="http://www.owl-
ontologies.com/Ontology1.owl#Sensor_Type"
/>
</owl:NamedIndividual>

<!-- http://www.owl
ontologies.com/Ontology1.owl#Chem2 -->
 <owl:NamedIndividual
rdf:about="http://www.owl-
ontologies.com/Ontology1.owl#Chem2">
   <rdf:type
rdf:resource="http://www.owl-
ontologies.com/Ontology1.owl#Chemical_pre
sence"/>
   <atTimeDate
rdf:datatype="&xsd;dateTimeStamp">2012-
05-13T09:00:03+01:00</atTimeDate>
   <hasChem
rdf:datatype="&xsd;boolean">true</hasChem
>
 </owl:NamedIndividual>
```

Listing 2: RDF Measure

```
<!--http://www.owl-
ontologies.com/Ontology1.owl#Detect_Explosi
ve -->

 <owl:Thing
rdf:about="&Ontology1;Detect_Explosive">
   <rdf:type
rdf:resource="&Ontology1;Detect_Event"/>
   <rdf:type
rdf:resource="&Ontology1;Explosive_Detectio
n"/>
   <rdf:type
rdf:resource="&Ontology1;Measurement"/>
   <rdf:type
rdf:resource="&Ontology1;Simple_Event"/>
   <rdf:type
rdf:resource="&owl;NamedIndividual"/>
   <Ontology1:is_Explosive_Detection
rdf:datatype="&xsd;boolean">true
</Ontology1:  is_Explosive_Detection>
   <Ontology1:Detect_by
rdf:resource="&Ontology1;CBRN1"/>
   <Ontology1:MeasureFrom
rdf:resource="&Ontology1;Chem2"/>
 </owl:Thing>

    <!--http://www.owl-
ontologies.com/Ontology1.owl#Detect_Possibl
e_Explosive -->

        <owl:Thing
rdf:about="&Ontology1;Detect_Possible_Explo
sive">
            <rdf:type
rdf:resource="&Ontology1;Composed_Event"/>
            <rdf:type
rdf:resource="&Ontology1;Detect_Event"/>
            <rdf:type
rdf:resource="&Ontology1;Possible_Explosive
"/>
            <rdf:type
rdf:resource="&Ontology1;Simple_Event"/>
            <rdf:type
rdf:resource="&owl;NamedIndividual"/>
            <Ontology1:Detect_Time
rdf:datatype="&xsd;dateTime">2012-05-
13T09:00:07+01:00 </Ontology1:Detect_Time>

<Ontology1:is_Possible_Explosive
rdf:datatype="&xsd;boolean">true</Ontology1
:is_Possible_Explosive>
            <Ontology1:Composed_From
rdf:resource="&Ontology1;Detect_Explosive"/
>
            <Ontology1:Composed_From
rdf:resource="&Ontology1;Detect_Intrusion"/
>
        </owl:Thing>
```

Listing 3: Simple and Composed Event

```
<!-- http://www.owl-
ontologies.com/Ontology1.owl#Allarme -->

 <owl:Thing rdf:about="&Ontology1;Allarme">
   <rdf:type
rdf:resource="&Ontology1;Alarm"/>
   <rdf:type
rdf:resource="&owl;NamedIndividual"/>
   <Ontology1:message
rdf:datatype="&xsd;string"></Ontology1:messag
e>
   <Ontology1:message
rdf:datatype="&xsd;string">Attention
Explosive Presence </Ontology1:message>
        <Ontology1335263048:Alarmfrom
rdf:resource="&Ontology1;Detect_Drop_Explosiv
e"/>
```

Listing 4: Alarm

REFERENCES

N. Konstantinou, E. Solidakis, A. Zafeiropoulos, P. Stathopoulos, N. Mitrou, 2010. A Context-aware Middleware for Real-Time Semantic Enrichment of Distributed Multimedia Metadata. *International Journal of Multimedia Tools and Applications* (MTAP), Springer, special issue on Data Semantics for Multimedia Systems, 46(2): 425-461.

Kaplanski, P., 2010. Description logic based generator of data centric applications. *Conference Proceedings of 2nd International Conference on Information Technology* (ICIT),. P 53-56. 2010. IEEE Publishing.

Liu, B., Ma, Y., Wong, C., 2000. Improving an association rule based classifier. *Journal of Principles of Data Mining and Knowledge Discovery.* P.P. 293-317.. Springer Verlag.

Wittig, R.D., Chow, P., 1996. OneChip: An FPGA processor with reconfigurable logic. *Conference Proceedings of IEEE Symposium on FPGAs for Custom Computing Machines P.* 126-135. 1996 IEEE Publishing

N. Konstantinou, E. Solidakis, S. Zoi, A. Zafeiropoulos, P. Stathopoulos, N. Mitrou, 2007. Priamos: A Middleware Architecture for Real Time Semantic Annotation of Context Features. *3rd IET International Conference on Intelligent Environments.*

Gomez, L., and Laube, A. 2009. Ontological Middleware for Dynamic Wireless Sensor Data Processing. In *Proceedings of the 2009 Third International Conference on Sensor Technologies and Applications* IEEE Computer Society, Washington, DC, USA, pp. 145-151.

Amato, F.,Casola, V., Gaglione, A., Mazzeo, A. 2011. A semantic enriched data model for sensor network interoperability. *Journal of Simulation Modelling Practice and Theory.* V 19. N 8. P 1745-1757.. Elsevier

V. Huang and M. Javed, *2008.* Semantic sensor information description and processing. *In 2nd International Conference on Sensor Technologies and Applications.*

Broekstra, J., Kampman, A., van Harmelen, F. 2002. Sesame: A generic architecture for storing and querying RDF and RDF Schema. *In ISWC.*

Horrocks, I., Patel-Schneider, P.F., Boley, H., Tabet, S., Grosof, B., & Dean, M. (May 2004). SWRL: A semantic web rule language combiningOWL and RuleML.

V. Casola, A. De Benedictis, A. Mazzeo and N. Mazzocca, 2011. SeNsIM-SEC: security in heterogeneous sensor networks, *SARSSI2011*

Prud'hommeaux, E., Seaborne, A., 2004. SPARQL Query Language for RDF.