A flexible network of sensors: case study

Anne-Marie Barthe-Delanoë
Université de Toulouse – Mines Albi
anne-marie.barthe@mines-albi.fr

Frédérick Bénaben
Université de Toulouse – Mines Albi
frederick.benaben@mines-albi.fr

Sébastien Truptil
Université de Toulouse – Mines Albi
sebastien.truptil@mines-albi.fr

Hervé Pingaud
Université Jean-François Champollion
herve.pingaud@mines-albi.fr

ABSTRACT

The goal of this article is to introduce a plastic architecture of a survey system dedicated to any kind of geographical area that requires to be observed. The principle of this architecture is to allow to change dynamically the set of sensors that is used to monitor the area and also to provide an analyze system able to deal with this unstable set of sensors. Based on Event-Driven Architecture (EDA) technology, such a system does not provide new features compared with traditional set of static sensors connected through cables to dozens of bulbs lighting when a predefined subset of measures is not in the expected range. However, the introduced architecture provides a completely agile and dynamic system of measurement where neither the network of sensors nor the system of measure interpretation is static.

Keywords

Event-driven architecture, Detection, Event pattern, Complex Event Processing

INTRODUCTION

When a crisis occurs, the aim of the crisis cell is to build and execute a response to solve (or at least reduce) the crisis situation. The MISE Project, as detailed in (Barthe-Delanoë, Bénaben, Carbonnel and Pingaud, 2012), proposes a Mediation Information System to help the stakeholders design and run collaborative response workflows and also a set of tools to detect the lack of accuracy between the given response and the crisis situation at time t. But these tools are dependent on the gathered knowledge about the crisis situation. Typically, in the case of an industrial crisis situation, as underlined by (Hahn, 2007), sensors are already localized on strategic places of the industrial plants and the way of the information is processed is determined and fixed. But, these sensors may be damaged, or not sufficient to cover the entire area concerned by the crisis: there is a need to add, remove, and switch sensors on demand in the survey network. There is also a need to adapt the way to exploit the gathered information.

CONTEXT

Sensor networks are historically physical networks: the sensors are connected to the central monitoring through wires. Such a kind of network is fixed: any change into the network structure or in the way to combine the information coming from various sensors is very hard to realize and very expensive. Even if since the nineties, wireless networks have emerged, the way to combine information gathered from sensors is not plastic, and the way to add new sensors into the network is not so easy (heterogeneity of protocols, of data format, etc.). Moreover, all industrial plants have not updated their existing sensor networks to wireless sensor networks, due to the cost of such an operation. This article focuses on how to gather and manage the information coming from various sensors, inside and outside the plant, in a plastic manner.

STATE OF THE ART

By nature and by the effects of the collaborative processes to solve or reduce the crisis, a crisis situation is an unstable and evolutionary phenomenon. So, as the crisis evolves, the information about the crisis characterization evolves too. It is crucial to (i) gather relevant information in real time and (ii) to manage it very quickly. Thus any changes, any evolution, any information that could challenged the crisis characterization accuracy have to be managed. According to (Chandy and Schulte, 2009; Etzion and Niblett, 2011; Luckham and Schulte, 2008) these elements that happened and that embedded data can be considered and managed as events.

The main hypothesis in this article is that the way the sensors communicate is SOA oriented (they have a web service interface, and if it not the case, creating such an interface is not a very hard point), which fits with the works of the Open Geospatial Consortium (OGC) on the definition of the Sensor Observation Service (SOS) (OGC, 2012) standard. This standard defines a web service interface to query sensor data and embedded a SOAP binding. The use of Event-Driven Architecture (EDA) principles (Michelson, 2006; Maréchaux, 2006) to complete the Service Oriented Architecture (SOA) principles allow us to take the events into account. A Complex Event Processing (CEP) engine is used to consume and manage these events. The amount of received events (especially in a crisis context) may be difficult to process by human beings as underlined by (Lagadec, 1993): people in charge of the crisis response may be information-saturated, which limits the management of events (filter events, deduce consequences from combined events, etc.). The CEP engine also filters and applies business rules to detect relevant events or combination of events in order to help the stakeholders to retrieve relevant information. At the implementation level, we need to decide what is the event format, the way to receive/emit events, and the CEP engine. (Luckham and Schulte, 2008) recommend defining the event types through one of the new computer languages like XML Schema or Java. It leads to use the Web Services-Notification (WS-N) standard (OASIS, 2007a; OASIS, 2007b; OASIS, 2007c), which is based on XML formalism, to describe the event occurrences sent by the sensors and received by their subscribers (including the CEP engine instance(s) dedicated to the considered crisis situation). The chosen CEP is Esper, developed by the American software editor EsperTech (EsperTech, 2013), as it natively proposes the management of XML formatted events.

EXAMPLE

The following example is based on a generic Chemical, Biological, Radiological, and Nuclear (CBRN) accident in an industrial plant (named Alpha). The plant map is shown on Figure 1: Alpha is located between a railroad, a road, a river and another industrial plant. An explosion occurred in Building B, where a volatile, water-soluble and unstable substance is stored. The substance is released outside its container. What are the contaminated areas? Where are they? How many contaminated areas at time t? What to do if the plant’s sensors are down? In a few words: *how to gather information about the crisis situation in a flexible way?*

Numerous survey sensors are located in various areas of the plant and its environment (as shown on Figure 1): depending on their role, they can measure air/water toxicity levels, and/or wind speed and direction. Moreover, the sensors are not necessary on Alpha’s site (like sensors #1 to #14): some of them are on the other plant site (#16 and #17), or are the property of authorities/institutions like the French national weather survey agency Meteo France or French local air survey organizations like Air Rhône Alpes or Air Normand (#18 and #19), or are not permanent or in a fixed location (e.g. mobile vehicle embedding sensors on board) such as sensor #15.

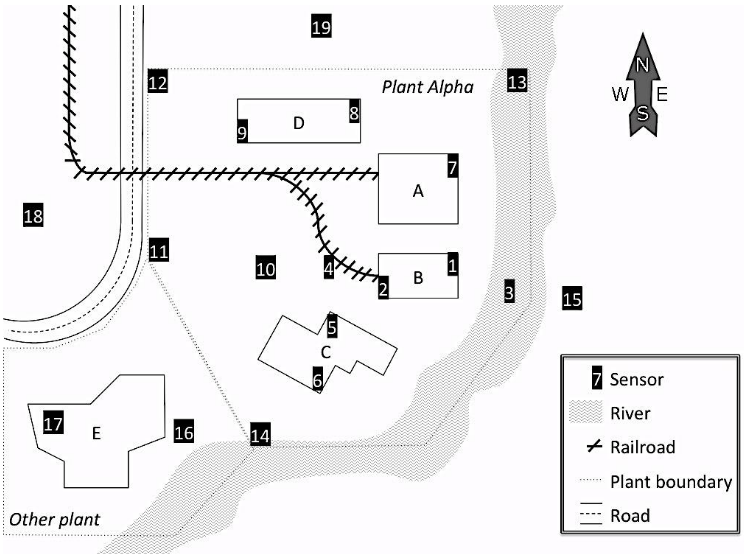

Figure 1. Map of plant Alpha and its sensors

The aim is to gather relevant information from these sensors, wherever they are, when needed, to characterize the crisis situation at any time.

SOLUTION

The existing sensor networks are not modified: a new virtual network is created, only on the base of the events emitted by the sensors. The web service interfaces of the sensors follow the WS-Notification standard to emit/receive events (it is implemented as an additional layer on the classical web service, and events are emitted/received on a different port from the one used by the web service endpoint). A CEP is provided and subscribes to the relevant event topics: it focuses on the kind of events it wants to listen, not on the providers of the events. The CEP engine can receive events emitted by various sensors, in any way that meets the subscribed event topics: events emitted by Alpha's sensors, the other plant's sensors, the national and local organizations' sensors, mobile and ephemeral sensors or even non conventional sensors (e.g. the user statuses of social networks like Facebook, Twitter, G+, or news feeds).

Business rules are human produced and then implemented into the CEP engine before it starts: they are triggered according to the events (or event patterns) the CEP receives. Through these rules, the CEP is able to create new events (alert events at least) and send them to the crisis situation monitoring to help the stakeholders to take a decision. For instance, it can help to define new survey areas, to decide to evacuate people (and where evacuate them), depending on the combination of measured levels of air/water toxicity and wind speed and direction (see Figure 2). It is interesting to note that the CEP is able to change its subscriptions to event topics on the fly.

In our example, at crisis breakdown, the first area to be surveyed is Area 1 (Figure 2). Only Alpha's sensors #1 and #2 provide information about the crisis situation. As the rate of toxic substance in ambient air inside Building B rises critical levels, a new survey area is defined around Building B: it's Area 2, including sensors #3 (in the river) and #4 in the virtual sensor network. But the rate of to toxic substance still increases in the air (data given by #4) and sensor #3 shows that the water of the river is contaminated. The surveyed area is expanded (Area 3) and covers now all Alpha plant. The sensors included in Area 3 are now event providers and allow surveying the entire plant. Unfortunately, the contamination of air and water is intensifying and the wind now blows to the South direction (this business rule is detailed below), decision was made to include the areas around Alpha plant into the survey area (Area 4). The added sensors (#15 to #19) are not owned or usually used by Alpha: they are the property of other organizations/institutions, but through the virtual sensor network, their emitted events can be shared with Alpha in order to help to gather knowledge about the crisis situation.

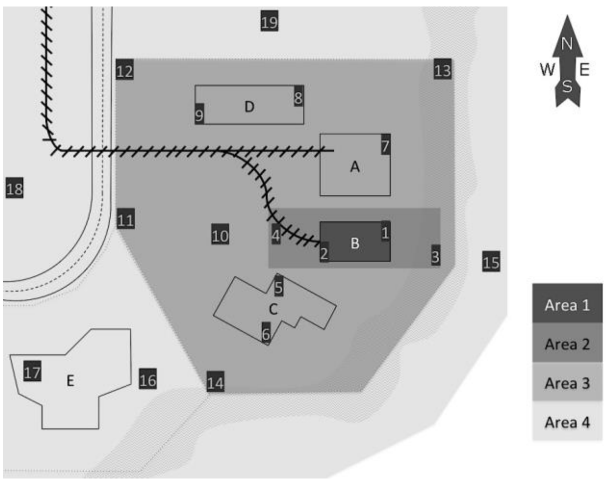

Figure 2. Map of survey areas and the used sensors in each area

In our research works, we developed connectors to allow Esper to subscribe and emit events following the WS-Notification standard. Web services, which also implement the WS-Notification standard, simulate the various sensors and emit events on the air quality survey, water quality survey, and wind survey. Business rules are defined and implemented into Esper. In our example, the aim is to define a dynamic survey area. For instance, if the average rates of toxic substance measured in water and in ambient air (during several time windows) and the wind speed (also during a time window) are over defined values, there is a strong contamination risk outside the current survey area.

A business rule is defined, combining the events sent by the air/water/wind sensors. The CEP applies this rule on the event streams. If the criteria are satisfied, the CEP advises stakeholders to expand the current survey area to the next defined area (cf. Figure 2):

If (average rate of toxic substance in water during the *last 2 minutes* is *over 0.5 g.L*$^{-1}$)
and (average rate of toxic substance in ambient air during the *last minute* is *over 600 µg.m*$^{-3}$)
and (wind speed in plant area is over 30km.h^{-1} during the *last minute*)
Then Emit_Alert("Contamination increasing. Advice: expand survey area")

We can note that in this rule (here written in natural language for a better understanding of the example), the sensors used to gather the measures of toxic substance rate have not to be namely known in the business rule. The only need is identify the relevant pieces of information (i.e. wind speed during the last two minutes, rates of toxic substance in water during the last two minutes, the rates of toxic substance in ambient air during the last minute) to get them through the events emitted by the sensors the CEP has subscribed. The sensor itself does not matter (as long it is supposed that the CEP subscribes to relevant event producers): the information is sensor-independent and topic-dependent.

Esper is launched and listens to the events emitted the sensors it has subscribed: on Figure 3, only two event streams are shown. They represent the streams of events concerning the values of the sensors' measures of toxic substance in ambient air and in water.

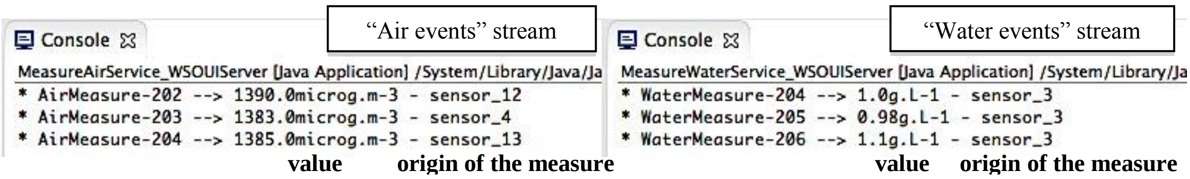

Figure 3. Screenshot of a part of the event streams generated by the sensors (Eclipse consoles)

When a pattern of events triggers a business rule, Esper creates and sends the stakeholders the associated event(s). As shown on Figure 4, the business rule detailed previously has been triggered: the CEP advises the stakeholders to expand the current survey area to Area 4. Based on the stakeholders' decision after having analyzed the events sent by the CEP, Esper can dynamically subscribe to new event topics to gather additional information through events emitted by the additional sensors. In the example, the stakeholders have decided to follow the advice given by the CEP. The new survey area is Area 4 and the new subscriptions focus on the sensors owned by Meteo France (#18 and #19), the ones owned by the Other Plant (#16 and #17) and a mobile unit (#15) provided by the local authorities in order to gather information about the environment out of Alpha's boundaries and combine them with Alpha's sensor data.

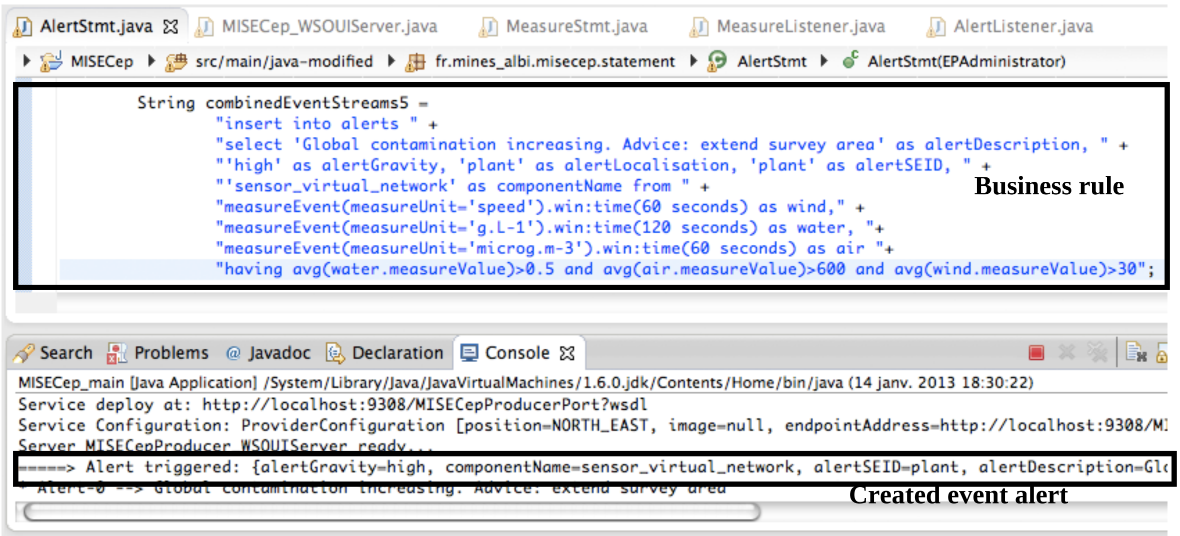

Figure 4. Screenshot of the developed application (in Eclipse), using Esper CEP engine and WS-N event subscriptions

CONCLUSION

In this paper we have shown that the implementation of a virtual plastic sensor network through an Even-Driven

Architecture and event subscriptions between independent systems can help to adapt the sensor network to the considered situation. This can be considered as an extension of the classical vision (commonly deployed in nuclear plants for instance) of a set of sensors connected to a monitoring table through static plugs and branches. However, in these research works, the set of sensors can be modified whenever and the system of measures interpretation remains valid with this dynamic network of sensors. This is mainly the consequence of using an Event-Driven Architecture where a CEP engine can apply business rules on an undetermined set of incoming events (data). Regarding that overall proposal, two reproaches might be done: (i) what about the robustness of such a system (considering how fragile are communication networks in crisis context)? And (ii) how to deal with the constraint of format of events, format of rules, etc.?

Regarding the first point, such an architecture is just as vulnerable as an other one: it is as strong as the physical system is strong. If the connection cables or the communication network is down, then the system is out. If there are redundancies in the branches, then the system is more robust. Etc.

Regarding the second point, this is mainly a question of equipping sensors with transmitters of events (in WS-Notification) so that they can provide their measure within the appropriate format. The definition of business rules (or patterns that should be identified) is a way to formalize the rough analysis that human beings are performing live when they are facing a large amount of incoming events. This would prevent crisis manager from being overwhelmed with information while ensuring them that they are observing the relevant area.

ACKNOWLEDGMENTS

This work has been partially funded by the French Research Agency (ANR) regarding the research project SocEDA (SOCial Event Driven Architecture) (Grant ANR-10-SEGI-013). This project aims to provide dynamic and adaptive workflows to collaborative situations (like crisis situations) through EDA and CEP. The authors would like to thank the project partners for their advices and comments regarding this work.

REFERENCES

1. Barthe-Delanoë, A.-M., Bénaben, F., Carbonnel, S. and Pingaud, H. (2012) Event-Driven Agility of Crisis Management Collaborative Processes, Proceedings of the 9th International ISCRAM Conference, Vancouver B.C., Canada.
2. Chandy, M. K., and Roy Schulte, W. (2009) Event Processing, Designing IT Systems for Agile Companies, McGraw Hill, USA.
3. EsperTech (2013), retrieved January, 2013 from: http://www.espertech.com/products/esper.php
4. Etzion, O. and Niblett, P. (2011) Event Processing in Action, Manning Publications, USA.
5. Hahn, D. (2007) Non-restrictive linking in wireless sensor networks for industrial risk management, Proceedings of the 4th International ISCRAM Conference, Delft, the Netherlands.
6. Lagadec, P. (1993) Preventing Chaos in a Crisis, Strategies for Prevention, Control and Damage Limitation, p.63, McGraw Hill, Maidenhead.
7. Luckham, D., and Schulte, R. (2008) Event Processing Glossary – Version 1.1. Event Processing Technical Society, retrieved March, 2011, from http://www.ep-ts.com/component/option,com_docman/task,doc_download/gid,66/Itemid,84/
8. Maréchaux, J.-L. (2006) Combining Service-Oriented Architecture and Event-Driven Architecture using an Enterprise Service Bus (Tech. Rep.), IBM Software Group, Canada.
9. Michelson, B. M. (2006) Event-Driven Architecture Overview: Event-Driven SOA Is Just Part of the EDA Story (Tech. Rep), Patricia Seybold Group, Boston.
10. OASIS (2006) Web Services Base Notification 1.3 OASIS Standard, retrieved January, 2012, from: http://docs.oasis-open.org/wsn/wsn-ws_base_notification-1.3-spec-os.pdf
11. OASIS (2006). Web Services Brokered Notification 1.3 OASIS Standard, retrieved January, 2012, from: http://docs.oasis-open.org/wsn/wsn-ws_brokered_notification-1.3-spec-os.pdf
12. OASIS (2006). Web Services Topics 1.3 OASIS Standard, retrieved January, 2012, from: http://docs.oasis-open.org/wsn/wsn-ws_topics-1.3-spec-os.pdf
13. OGC Open Geospatial Consortium (2012). OGC Sensor Observation Service Interface Standard, retrieved February, 2013, from https://portal.opengeospatial.org/files/?artifact_id=47599