Demonstration: Defining and Detecting Complex Events in Sensor Networks

Lucas Leidinger and Kerry Taylor

CSIRO ICT Centre
GPO Box 664, Canberra, ACT, 2602, Australia
kerry.taylor@csiro.au
lucas.leidinger@gmx.de

1 Introduction

Improvements in technology and manufacturing reduce prices for sensing technology and allow the internet-connection of more and more base stations; therefore growing both the scale and application areas for wireless sensor networks. Reconfigurable, general purpose networks are replacing classic black-box sensor technology in the fields of environmental observations, for example conservation and disaster prevention, production, logistics, medicine and security.

Such networks provide high volumes of continuously captured measurement data. However, often only a small amount is interesting or has relevance for domain applications. Certain information may be required by different client applications, at several locations at the same time. The installation and configuration of sensor networks can quickly become complex, especially, if platforms and sensors of various manufacturers with different setups have to form a common network.

This demonstration deals with the idea of using semantic technologies and Complex Event Processing (CEP) to define and to detect complex events arising in the data collected by heterogeneous wireless sensor networks. A complex event has to be understood as a combination of filtered measurement values from particular sensors and locations in a well defined order within a specific period of time. The problem of programming the sensors and configuring the CEP system in several different low-level programming languages will be abstracted. The use of semantic ontologies allows the definition and detection of complex events independent from the type of sensor or kind of CEP system. For this purpose, event definitions, sensor programs and CEP queries are modelled in a custom ontology. The use of dynamic ontology assertions allows the recognition and reuse of existing sensor programs and available data streams. It makes it possible to perform semantic optimizations and to dynamically build a user control interface.

2 Architecture

To prove the idea of *ontology-driven complex event processing in heterogeneous sensor networks* and for demonstrating the use of semantic technologies combined with sensor networks, an *Event Framework* was developed.

An OWL2 *EventOntology* is the central part of the entire *Event Framework*. It is designed to store definitions of complex events and allows the use of reasoning and classification over event information to obtain additional knowledge and to perform semantic optimizations. Semantic optimizations for both sensor node programs and data streams are applied: ontology definitions of existing sensor programs and CEP streams are checked (by concept subsumption in the first case and concept membership in the latter) prior to creating new ones. If suitable pre-existing concepts are found in the ontology then the corresponding pre-existing sensor and instrument resources and CEP stream configurations, respectively, can be reused: saving instrument resources and reducing the amount of data which must be transferred between data source and event processing application. Hermit is used for the reasoning services.

The *User Interface*, an extension of the popular OWL ontology editor *Protégé*, allows one to define events in a logical and expressive way and to store this definition in an ontology. The entire complex event definition is composed of different parts: *Events*, *Alerts*, *Observations*, *Triggers* and *Sensor Programs*.

Semantic Event Middleware is the counterpart to the user interface. All created complex event descriptions which have been transformed into ontology data are forwarded from the *User Interface* to the *Semantic Event Middleware*. Here, the reverse process to transforming a user description into ontology data is performed.

To abstract the specification and access of event data streams, a *Management Module Interface* has been designed which allows the implementation of an independent solution for each specific kind of instrument or event source. Each distinct sensor network technology requires an implementation of the interface as a wrapper over the heterogenous aspects of the sensor network that are not modelled in the ontology (such as the native programming language grammar and the compilation/downloading tools).

The (*Coral8*[1] CEP-platform performs the actual complex event detection. For this, the server receives a stream with information about the event data sources and a query which contains the program for the complex event detection. The CEP server is also responsible for sending the user defined alert message if an event has been detected.

3 Demonstration

The demonstration will show the interaction of the individual components of the *Event Framework*. While some sensor data will be simulated, we plan to also

[1] The Coral8 CEP-platform. `http://www.aleri.com/products/aleri-cep/coral8-engine`

use a live Environdata WeatherMaster1600 instrument to make environmental observations and to show the *Event Framework*:

1. Showing the general functionality of the *User Interface*.
2. Defining a complex event definition composed from several atomic sensed events within the *User Interface*.
3. Processing the complex event definition: transformation into ontology data, programming selected instruments and start of the CEP event detection.
4. Simulating sensor data and changes in the environmental observation by the WeatherMaster1600 instrument.
5. Showing the detection of the defined event and the consequent alert.

4 Conclusion

The sensor programming function and it's ontology modelling allows high-level programming for sensor instruments, and can be used quite independently of the event detection function[3]. The event processing capability is described in more detail here[2].

The *Event Framework* is now being upgraded to work with an alternative data stream management system (GSN[1]) and will be deployed in February 2012, over a network of soil moisture sensors and smart ear-tagged cattle on a demonstration farm near Armidale, New South Wales, Australia. In this case it will be used as part of a system for precision agriculture: to alert the farmer to issues associated with livestock management (e.g. herd location, herd state), pasture management (e.g. plant water availability, pasture yield estimates), and joint management (e.g. that a herd should be moved from one paddock to another).

References

1. Karl Aberer, Manfred Hauswirth, and Ali Salehi. A middleware for fast and flexible sensor network deployment. In *Proceedings of the 32nd international conference on Very large data bases*, VLDB '06, pages 1199–1202. VLDB Endowment, 2006.
2. Kerry Taylor and Lucas Leidinger. Ontology-driven complex event processing in heterogeneous sensor networks. In *The Semantic Web: Research and Applications: Proc. 8th Extended Semantic Web Conference*, volume 2 of *LNCS 6644*, pages 285–299, Heraklion, Greece, 29th May to 2nd June 2nd 2011. Springer.
3. Kerry Taylor and Patrick Penkala. Using explicit semantic representations for user programming of sensor devices. In *Advances in Ontologies: Proceedings of the Australasian Ontology Workshop*, volume 112 of *Conferences in Research and Practice in Information Technology*, Melbourne, Australia, 1 December 2009. Australasian Computer Society.