Ontology Driven Adaptive Data Processing In Wireless Sensor Networks*

Work-in-progress

Yuheng Hu[1] Zhendong Wu[2] Ming Guo[1]†

[1] School of Computing, Zhejiang University City College
[2] College of Computer Science, Zhejiang University

ABSTRACT

It is important to provide adaptive data processing in wireless sensor networks in order to deal with various applications. In this paper,we propose a WIreless Sensor Networks Ontology (WISNO) for flexible modeling of sensor data. WISNO contains two-tier ontologies, a front-end for coarse-grained analysis and a back-end for high-level fine-grained data processing. We also describes the WISNO reasoning rules that adopts description logic and SWRL for managing data automatically.

Categories and Subject Descriptors

C.2.3 [**Computer-Communication Networks**]: Network Operations—*Network management*

General Terms

Management, Performance

Keywords

Wireless Sensor Networks, Ontology, SWRL, description logic

1. INTRODUCTION

With the development of sensor networks, a adaptive strategy for sensor-data processing and management is needed. Ontology is widely used to enrich data description, which facilitate data processing and management. Most of the current research focus on how to use the single ontology to model data and load the sensor data into knowledge base for subsequent application. However, merging huge volume of sensor data and deliver them to the end-user would cause high communication cost and processing cost.

We propose a two-tier ontology driven framework WISNO for flexible data processing in wireless sensor networks. In this framework, the front-end performs lightweight reasoning at the sensor nodes for coarse-grained analysis and regulating data stream. This fits well with the resource limited sensor nodes. The back-end performs deep data analysis at high-performance servers. We adopt Description Logic to enable context classification and comparison at the front-end, while SWRL [1] is used to encode specific rules according to different context of spots.

*This work is supported by National Science Foundation of China under Grant No.60473025 and Zhejiang Provincial Natural Science Foundation of China under Grant No.Y106427

†Correspondent author. Email: guom@zucc.edu.cn

Infoscale'07 June 6-8, 2007, Suzhou, China

2. APPLIED SCENARIO

We propose our scenario (Figure 1) that using WISNO to merge and manage the data. Specifically, the front-end is responsible for capturing environmental data and coarse-grain analysis, whereas the back-end takes charge of further data analysis by heavyweight reasoning and issuing high-level actions.

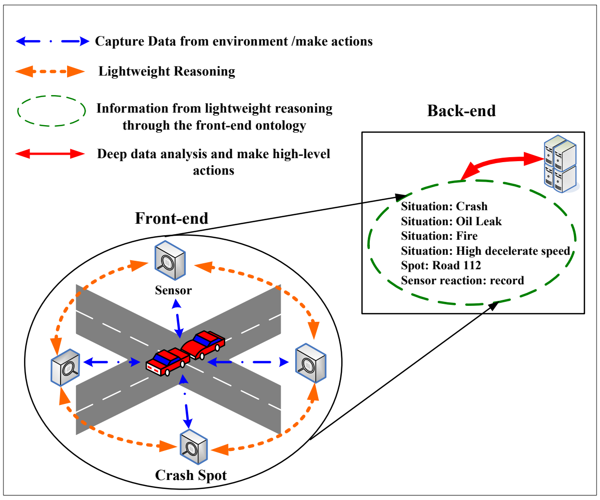

Figure 1: Crash demo in WSN

3. THE PROPOSED ONTOLOGY

WISNO is a Semantic Web compatible ontology developed using Protege[1]. Currently, the front-end ontology (Figure 2) provides some fundamental concepts, e.g. sensor, location, and spot situation as well as a set of sub-classes to collect basic features of environmental context.

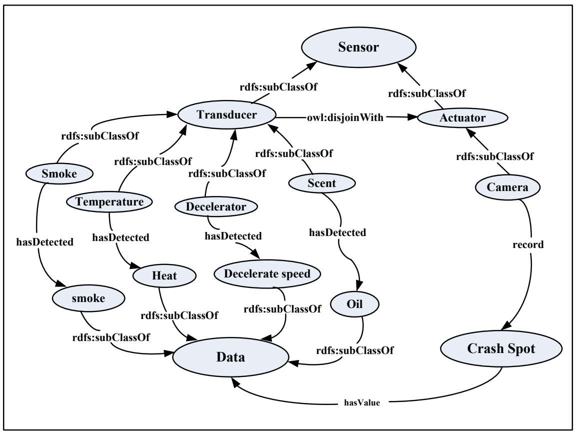

Figure 2: Partial definition of specific ontology for Front-end

The back-end (Figure 3) contains the front-end concept imported by <owl:imports> syntax and collection of detailed sensor properties, e.g. sensor energy capacity and sensor state to improve precision of reasoning.

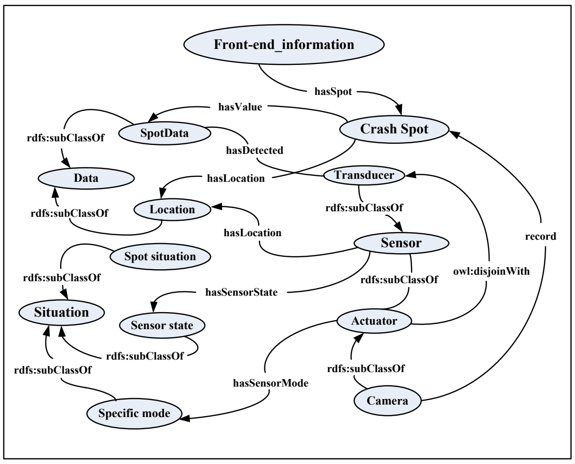

Figure 3: Partial definition of specific ontology for Back-end

WISNO deviates from SensorML [2] since it lacks the semantic richness for inference. In addition, we extracted the commonly used terms in sensor domain from IEEE 1454.1 smart transducers template description language[3].

4. CONTEXT REASONING

The context reasoning in WISNO has two main usages: deducing high-level, explicit information from low-level, implicit context, and checking the consistency of ontologies. Sensor data and properties are treated as instances of cocepts defined in front-end or back-end.

4.1 Front-end Reasoning

The mission of front-end reasoning is to evaluate data and determine whether it needs to be send for further analysis in back-end.

We adopt DL as the foundation of inference. DL approaches integrate powerful inference mechanisms to reason on the schema, the reasoner computes the concepts instances belong to by determining whether instances statisfy the constraint.

Table 1 shows a DL rule related to the crash scenario depicted in Section 3. We assume a crash by: 1)high decelerate speed with 2) oil leakage on road and 3)fire in the spot.

Table 1: Front-end rules for situation

Front-end Reasoning Rules
Crash $\equiv \exists$ has_Detected.Oil_Leak $\sqcap \exists$ has_Detected.Fire $\sqcap$ $\exists$ has_Detected.Decelerate_Speed

4.2 Back-end Reasoning

In back-end reasoning, we load additional sensor properties, (e.g location, states) with data from front-end into heavyweight inference for deep data analysis and sensor controls.

Table 2 shows one of the rules for getting and setting sensor state: the camera sensor changes its working mode to infrared mode when fire detected.

Table 2: Back-end heavyweight reasoning rules

Back-end Reasoning Rules
(?camera *getSensorState* working)$\wedge$ (?spot *has_spot* Fire) $\Rightarrow$(?camera *setSensorMode* Infred_Mode)

5. CONCLUSION AND FUTURE WORK

Our study in this paper shows the WISNO framework is feasible for supporting adaptive data-processing in WSN. Currently, we are working on implementing a prototype using Pellet[2] to reason about ontologies and Jess[3] for reasoning about rules, both accessed through Jena[4] API. As for future work, we plan to test effectiveness of WISNO by quantitatively measuring the reaction speed in a real environment.

6. REFERENCES

[1] I.Horrocks et al., SWRL: A Semantic Web Rule Language Combining OWL and RuleML, W3C Member Submission, 2004

[2] M. Smith. Sensor Model Language (SensorML) for In-situ and Remote Sensors. OpenGIS Consortium, Inc., December 2002.

[3] C. H. Jones. IEEE 1451.4 smart transducers template description language, http://standards.ieee.org/regauth/1451 accessed/Tutorials.html, April22, 2006.

[1]http://protege.stanford.edu/

[2]http://www.mindswap.org/2003/pellet

[3]http://herzberg.ca.sandia.gov/jess

[4]http://jena.sourceforge.net/