Bridging the Gap between Semantic Web and Networked Sensors: A Position Paper

Xiang Su and Jukka Riekki

Intelligent Systems Group and Infotech Oulu,
FIN-90014, University of Oulu, Finland
{Xiang.Su,Jukka.Riekki}@ee.oulu.fi

Abstract. In this position paper, we present our work towards designing a Semantic Web languages-compatible representation for networked sensors. The representation, Entity Notation, is proposed to connect sensors to the Semantic Web. Entity Notation can express RDF and OWL ontology models in a uniform format. Meanwhile, it offers a lightweight alternative for sensors with limited computation and communication capabilities. We present motivation and design issues of Entity Notation in this paper.

Key words: Entity Notation, Semantic Web Representations, Embedded Sensing

1 Introduction

Sensors and sensor networks will be everywhere, from environment monitoring to planet exploration. Increasing maturity of sensors and sensing technology opens up questions about how to effectively operate with massive amounts of data produced by current large scale deployments. What kind of solutions should be developed on top of these sensor networks to make the best use of all the data?

Semantic Web approaches this challenge from the opposite direction and offers solutions for a heterogeneous, large scale inter-operating networks. When sensors meet Semantic Web, there will be benefits for both sides. For the Semantic Web, sensors and sensor networks bring a whole new peripheral of massive data from real time measurements. For sensing systems, Semantic Web enables resource sharing, resource reallocation, and advanced functionality like inference over events in space and time within a particular context.

One main challenge for connecting networked sensors to the Semantic Web is that most sensors produce raw data in their own formats. Hence, each sensor model needs its own solution when sensors are connected to Semantic Web. We tackle the gap between Semantic Web and networked sensors at the data interchange level. As RDF [1] and ontology [2] models are based on triplet representations, we design a uniform format, Entity Notation (EN), based on triplets for expressing RDF models and OWL ontologies. Moreover, we suggest a compact format for transmission when resources of sensor networks are constrained. The compact format shortens our representation based on techniques of templates and prefixes. Entity Notation is unique in its combination of expressiveness and compactness. This combination makes it ideal for connecting networked sensors to the Semantic Web.

We start this paper by introducing related data representations in section 2. Then, we present our requirements and design details in section 3. Finally, we discuss our future work in section 4.

2 Related Work

Numerous data representations have been defined in different sensing systems, like Generic Sensor Format [3], Unisens [4] and Unified Transportation Sensor Data Format [5]. Sensor model specific data formats introduce dependencies into the system, and even set up a barrier to web applications. Binary XML emerges as a new trend to support convenient solutions for transferring raw data to XML representation. There are many competitive binary XML formats, like X.694 [6], Efficient XML Interchange Format (EXI) [7], and Fast Infoset [8]. Data in these formats can be transformed into XML straightforwardly, and some of them can achieve good compression ratio. However, XML representation has no embedded semantics, and the gap between syntax and semantics still exists.

On the other hand, there are some compact serializations for representing knowledge, like N3 [9] and Turtle [10], which can express RDF in a straightforward manner. They are flexible languages with strong expressive capability. But unfortunately, they normally contain shorthand syntaxes to shorten the files. This is not a suitable solution for resource-constrained sensors, because they do not have enough processing power to process these shorthand syntaxes.

Another effort to increase interoperability is to annotate sensor data with semantic metadata. Metadata languages, like RDFa, are utilized to provide context information for raw data [11]. However, as the expressivity of annotation formats is limited, they cannot capture the full semantics of sensor data. In a nutshell, more work is needed in developing representation that can interoperate Semantic Web knowledge with sensing data straightforwardly and unambiguously.

3 Designing Entity Notation

Our ultimate goal is to design a communication-convenient serialization for Semantic Web languages. This serialization will ease the transferring of knowledge models and advance interoperability between Semantic Web and networked sensors.

We examine W3C Semantic Web standards, and consider RDF and OWL ontology as essential models for realizing knowledge interoperability. Intelligent Semantic Web applications normally build on data represented by RDF and knowledge based on ontologies. Then inference tools can be deployed on this knowledge base to enable intelligent functions. At current stage, EN can serialize RDF models and OWL DL ontology in a uniform format, and we will extend this format for other standard languages.

3.1 Design Requirements

We have specified the following main design requirements for EN. First, EN must be expressive enough for representing RDF and OWL DL ontology models in a uniform format. It should be possible to transform any description of RDF and OWL DL knowledge into EN packets, and vice versa. Fulfilling this goal enables transforming RDF/XML and OWL into EN packets. This is essential, as XML is one of the most widely used serialization, and OWL utilizes XML syntax too. Many available reasoning tools are based on RDF/XML and OWL. Second, EN must support distributed knowledge production. Any snippet of RDF and ontology knowledge can be defined by any device and Semantic Web applications, and these snippets can be discovered, transferred and integrated into larger ontologies. This will be a unique feature that to the bests of our knowledge has not been reported before. Third, EN must be suitable for bandwidth limited and ultra-low power communication links of sensor networks. For example, an unsophisticated embedded sensor should be able to compose EN packets using minimal computation power and deliver packets to Semantic Web applications in an energy efficient manner. Fourth, EN must be able to carry any kind of information between sensors and Semantic Web applications. Fifth, the information carried in EN packets needs to be identified in a unique fashion. This is an important feature for an Internet scale system when all the possible usages of the information cannot be specified in advance.

3.2 General EN Packet Formats

The basis of RDF models is a sequence of triplets (*Subject*, *Predicate*, *Object*), and ontologies also contain simple definitions and triples, which describe relations among concepts. Complex relations are built on simple definitions and triplets. Besides, all information is identified in a unique fashion by URIs, except literals.

To fulfill the expressibility and lightweight requirements, we define two EN formats: the complete format, and the short format. Complete format has enough expressibility, as it follows the triple notation. Moreover, we adopt almost all terms from OWL in the complete format. It's straightforward to serialize RDF and ontologies into complete EN packets and vice versa. Lightweight short packets can support resource-constrained devices and slow communication links. Furthermore, EN supports composition and decomposition of knowledge, and thus enables incremental knowledge definition and communication. We define for RDF and ontology knowledge a uniform EN format that does not constrain the type of information. Finally, EN uses UUIDs (Universally Unique Identifiers) [12] and URIs for unique identification.

Generally, an EN packet depicts an entity and its relationships with values and other entities. An entity is some identifiable whole, concrete or abstract. Entities in the RDF level knowledge tend to be concrete, like sensors, robots and measurements. Entities in the ontology level are abstract concepts and relations in the ontology. An entity description is of the form:

```
EntityType EntityID
 PropertyName PropertyValue
 ...
 PropertyName PropertyValue
```

The subject can be same for multiple triplets, and this happens for most entities in actual knowledge models. Hence, we identify one entity in an entity description, and then any number of triplets can be defined about it. Besides, we include type information for EntityID, because this is important information to identify its subsumption.

Square brackets and angle brackets are utilized to identify the level of knowledge an EN packet should mapped to. When an entity description is wrapped in square brackets ([and]), this EN packet should be transferred to RDF models. When an entity description is wrapped in angle brackets (< and >), this EN packet should be transferred to OWL ontologies.

When an EN packet is mapped to RDF statements, EntityID maps to the subject of a triplet, PropertyNames map to predicates, PropertyValues map to Objects, and EntityType maps to a class this entity is subsumed. When it should be mapped to ontology snippets, EntityID maps to an element name, EntityType indicates the type of the element, and PropertyNames are mapped to relationships or EntityCharacteristics. Relationships between entities, like rdfs:subClassOf, and rdfs:domain, can be mapped to OWL constructors. EntityCharacteristics can be mapped to the term rdf:type, while CharacteristicsName can be mapped directly to OWL types.

For resource-constrained sensors, complete EN packets are still verbose as full URIs are used. We do not utilize complex compression algorithms, because this would introduce additional resource-consumption for sensors. Instead, we suggest a short packet format for decreasing the need for computations and communication bandwidth.

The short EN format uses templates and prefixes to shorten packets. The basic idea is that a template contains a description of the constant part of a sequence of EN packets and placeholders for the variable items. The packet sent over the communication links needs to contain only a template identifier and the variable items. A sequence of complete EN packets can then be assembled by replacing the template's placeholders with the values contained in a short packet. Prefixes are used to shorten URI references. The receiver, like a gateway, needs to possess the corresponding template in order to assemble complete packets but a resource-constrained sensor needs only the template identifier. Prefixes are used to shorten URI references. Similar to complete packets, square or angle brackets are utilized to wrap following descriptions depending on the level of knowledge:

```
UUID PropertyValue  ... PropertyValue
```

UUID is an identifier that is guaranteed to be unique across space and time. In its canonical form, a UUID is 128 bits long. We utilize UUIDs to identify templates. When a short packet format includes EntityIDs the format is:

```
UUID
EntityID PropertyValue ... PropertyValue
    ......
EntityID PropertyValue ... PropertyValue
```

3.3 The Role of EN in Connecting Semantic Web and Networked Sensors

Figure 1 presents the role of EN in bridging the gap of Semantic Web and networked sensors. Embedded sensors can compose complete or short EN packets depending on their capabilities, but they all can transformed into Semantic Web representation at gateways. Knowledge base and Semantic Web applications can access sensor produced data and knowledge straightforwardly. Moreover, some devices in sensor networks can actuate commands what Semantic Web applications produce.

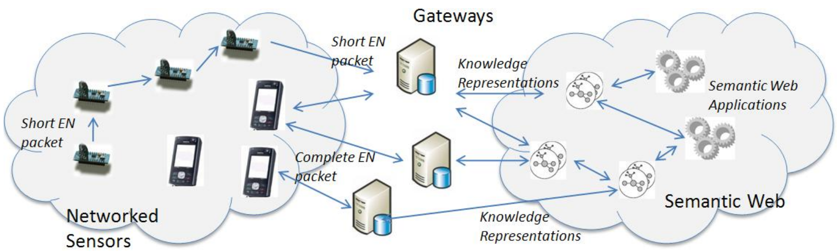

Fig. 1. The role of EN in bridging the gap of semantic web and networked sensors

4 Discussion

In this short paper, we presented motivation of designing a novel representation for connecting networked sensors to Semantic Web. Then, we introduced design requirements, general formats of EN, and EN's role in connecting networked sensors to Semantic Web. We have implemented EN in a simulator, wearable sensors, and embedded GPS sensors of Nokia N95 cell phones. Implementation details are not included in this paper.

EN is unique in its combination of expressiveness and compactness. Utilization of UUIDs and URIs enables EN a scalable solution for Semantic Web. It is recommended to adopt full 128 bits UUIDs for web-scale system. The shorter alternatives, like 16 bit, 32 bit UUIDs, guarantee uniqueness in smaller scale systems. UUIDs offer a convenient solution for mapping shorter UUIDs to full ones, which facilities the integration of small scale systems to large, even Internet scale systems.

The current EN version does not support nesting although nesting is widely supported by data representations and it compresses descriptions. Instead of nesting we divide descriptions into several small EN packets and use URI references and blank node identifiers to determine relations between them. Small packets facilitate optimizing the communication. For example, packets can have different routing paths or they can be sent over a longer period of time. We will consider nesting in near future; we will analyze the computational load it sets for constrained devices and decide based on this analysis whether EN should support nesting.

This is a early work, and we will continue it with several aspects. We will specify formal syntax and semantics for our serializtion, and perform more experiments based on standard dataset to evaluate it. Moreover, we will study how to represent SWRL and RIF rules in EN. This is another important step for building a uniform format for transferring representations of Semantic Web. Finally, we will explore possibilities of applying EN and Semantic Web technologies in sensor networks, especially how to infer knowledge from sensory data in ubiquitous environments.

Acknowledgments. This work was funded by Infotech Oulu and the MOTIVE research project program of the Academy of Finland.

References

1. RDF Primer, `http://www.w3.org/TR/rdf-primer/`
2. Studer, R., Benjamins, V.R., Fensel, D.: Knowledge Engineering:Principles and methods. Data Knowledge Engineering. 25, 161–197 (1998)
3. Generic Sensor Fromat (GSF), `http://www.saic.com/maritime/gsf/`
4. Unisens, `http://www.unisens.org/index.php`
5. Kwon, T.M., Dhruv, N.: Unified Transportation Sensor Data Format (UTSDF) for Efficient Archiving and Sharing of Statewide Transportation Sensor Data. In: The Transportation Research Board 83rd Annual Meeting, Washington D.C. (2004)
6. International Telecommunication Union, X.694, `http://www.itu.int/ITU-T/studygroups/com17/languages/X694.pdf`
7. W3C Candidate Recommendation, Efficient XML Interchange (EXI) Format 1.0, `http://www.w3.org/TR/exi/`
8. Fast Infoset Binary XML for .Net, `http://www.noemax.com/products/fastinfoset/index.html`
9. Notation 3 (N3) A readable RDF syntax, `http://www.w3.org/DesignIssues/Notation3`
10. Turtle - Terse RDF Triple Language, `http://www.w3.org/TeamSubmission/turtle/`
11. Sheth A., Henson C., Sahoo S.S.: Semantic Sensor Web. IEEE Internet Computing. 1089-7801/08, 78–83 (2008)
12. A Universally Unique IDentifier (UUID) URN Namespace, `http://www.ietf.org/rfc/rfc4122.txt`