

Ubiquity Symposium

The Internet of Things

A Case for Interoperable IoT Sensor Data and Meta-data Formats

by Milan Milenkovic

Editor's Introduction

While much attention has been focused on building sensing systems and backing cloud infrastructure in the Internet of things/Web of things (IoT/WoT) community, enabling third-party applications and services that can operate across domains and across devices has not been given much consideration. The challenge for the community is to devise standards and practices that enable integration of data from sensors across devices, users, and domains to enable new types of applications and services that facilitate much more comprehensive understanding and quantitative insights into the world around us.

The target is to achieve semantic or, as the Industrial Internet Consortium (<http://www.iiconsortium.org/IIRA.htm>) refers to it, “conceptual interoperability” i.e., represent information in a form whose meaning is independent of the application generating or using it. When supported, semantic interoperability achieves two important objectives: (1) It enables service-level integration of IoT end-to-end systems constructed using components from different vendors, such as a variety gateways running different middleware, with a third-party cloud for data storage, processed by analytics from independent vendors; and (2) It allows aggregation of data from different domains, such as disparate systems in smart cities, to allow for holistic management and—more importantly—to enable big data by virtue of creation of large data sets that are understandable, and thus usable, to analytics and other services.

Useful instantiations of multi-domain IoT systems can provide significant new business and innovation opportunities for new services. In order to fulfill that promise, IoT/WoT systems need to be designed to support some level of commonality by defining interoperable sensor data and meta-data formats, naming, taxonomy and possibly ontology. This paper presents use cases that motivate this need and outlines a possible path to get there.

Ubiquity Symposium

The Internet of Things

A Case for Interoperable IoT Sensor Data and Meta-data Formats

by Milan Milenkovic

A simple but powerful view of the Internet (web) of Things (IoT/WoT) is as a sensor-enhanced Internet, i.e., sensors attached to the Internet, directly or via intermediaries, with the ability to source data and, where appropriate, to provide actuation and possibly physically impact the real world. Connected sensors share real-world data virtually. This seemingly simple addition is a transformational change; it basically bridges the gap between physical and virtual/cyber worlds that has persisted since the invention of modern computing. In effect, the IoT is the Internet with all of its features and capabilities with the addition of a real-world dimension (and interface). As has been observed, the Internet becomes a web of people, information, services, and things. This view of IoT/WoT is somewhat different from M2M (machine to machine) systems, which focus primarily on machines talking to each other over available connections, including the Internet. Internet of everything is more inclusive in the sense that it assumes everything is connected to everything else using Internet fabric and protocols, and thus has the potential to engage in interactions and to provide a plethora of exciting new uses and services limited only by creativity and security/privacy restrictions.

Sensors usually convey information about real-world phenomena, widely ranging from direct measurements, e.g., temperature, to user observations, e.g., an overflowing riverbank. Our definition of sensors is very broad; it includes not only all types of hardware sensors and sensor networks, but also software sensors, sensing services, and people. Software sensors are usually software agents that can capture and report on some real-world condition of interest, such as user presence detected via key clicks or mouse movements. Sensing service refers to data provided by an external source with programmatic interfaces, e.g., a localized weather reporting service. The concept of people as sensors refers to users providing direct input—their observations, comfort, or system adjustment preferences—via dedicated end-user interfaces or social networks.

Opportunity

At present, much of this sensor data is of limited value as it is locked in closed systems and used by proprietary usage-specific and device-specific applications. A great opportunity lies in being able to create horizontal services and applications that make use of aggregations of sensor data across devices and domains. Such services can better serve interests and needs of users, e.g., as data from devices belonging to a user or of interest to the user in a particular context, surrounding, or at a given point in time. An example is being able to discover sensors of interest in user's proximity, such as discovering nearby clusters of air-quality sensors in a smart city and visualizing that information on a personal smart device. If the user asks, "What is the air quality where I am right now?", the device will access nearby sensor data. Let's take it a step further, a car can register vibrations when going over a pothole and transmit related GPS coordinates to inform the city's maintenance department of road conditions. This is crowdsourcing that is timely, accurate, and saves labor costs. Being able to correlate data across usually isolated vertical systems results in improved efficiency, for example tracking building occupancy in real time and proactively adjusting to load variations—such as when a sizable number of people go out for lunch on a nice day, or when they leave earlier than usual prior to a holiday. Even these relatively simple examples highlight potential benefits of linking sensing data with services across domains, such as individuals, buildings, neighborhoods, and cities.

[McKinsey estimates](#) achieving interoperability in IoT would unlock an additional 40 percent of the total market value, which they estimate will be \$11.1 trillion in 2025. Alternatively, failure to interoperate may reduce IoT total possible market by 40 percent.

Sensor data aggregation is a pre-condition for harvesting big data and analytics, but aggregate data can be useful only if it can be read, understood, and placed in context by services that make use of them. In other words, interoperability of sensor data and metadata formats is essential for realizing the big-data potential in IoT.

As an illustration, consider a scenario of creating an application that tracks a person's energy footprint. It could start by aggregating data on that person's energy usage at the office, at home, and during their commute, possibly offset by tracking energy-saving activities such as walking or biking to work. Later on, tracking the carbon impact of long-distance travel could be added to complete the picture. And to make it more valuable, the system could provide comparisons with averages relevant to users across aggregations of interest—such as footprints and usage range of people in the same company, neighborhood, city, country, and, ultimately,

the world. This could be used for better insights, to train user intuition, and possibly to enable competitions or tracking of group energy saving goals in social circles of interest.

Such an application or service would need to use data from a variety of sensors attached to legacy systems like BMS (building management system), IoT-aware instrumented systems like home automation and vehicles, or a personal or user-targeted sensor like an office power-strip energy meter and wearable activity tracker. With today's state of affairs, this would be cost prohibitive due to expensive custom coding, or even impossible as sensor data are locked in fragmented and often proprietary vertical silos, including, in this example, BMS, vehicle information system, home automation, energy meters, and wearable fitness/activity tracker. A streamlined, web-style approach is to acquire sensor data to be translated into interoperable data and meta-data formats, and then deposited into aggregation pools for processing by a variety of services and applications.

Useful elaborations of multi-domain IoT systems provide significant new business opportunities for services, big data, and analytics. In order to fulfill that promise, IoT systems need to be designed to support data interoperability and achieve, at least, some level of commonality in taxonomies, ontologies, naming, and meta-data assignment and processing.

A Path to Get There

With explosive growth of the IoT, there is not enough time to engage in a traditional, and usually lengthy, standards setting activity. With the current course and speed, it is quite likely that no single data and metadata format will emerge and be adopted by traditionally independent and uncoordinated domains—such as building automation, transportation, or energy management. Instead, or in the interim, we advocate a minimalistic pragmatic approach that is “good enough.”

Interoperability does not imply use of common data and meta-data formats in different systems—an almost impossible feat in IoT given the proliferation of standards and proprietary formats in this space. Interoperability does require sufficient specification and adherence to some basic design principles. This in turn enables interoperability. In this case, the ability to perform (automated) machine translation of data and meta-data formats across systems, domains, and—where desired—individual system components and devices.

It should be pointed out data interoperability does not imply free sharing. Data and meta-data interoperability is a necessary, but insufficient, condition for sharing. Decisions as to what to share and with who are subject to business and legal arrangements between data owners and

users, and should be enforced by security and privacy constraints imposed on the data even when stored in aggregated pools.

Herein we focus on service-level data interoperability. Data can be acquired in their respective domains and encoded into an interoperable format at any point up to, and including, the API service call. Early canonical encoding may provide additional benefits, such as device-level interoperability, which is a sufficient, but not necessary, condition for service-level data interoperability.

Meta-data, also referred to as tags, annotates data reported by sensors to provide context. Some examples of meta-data are sensor type, function, location, manufacturer, and serial number. The primary function is to provide contextual semantics to create “rich data” (basically data made useful) for a variety of post-processing services and applications, such as analytics and device/asset management. It also enables searches of annotated data by attribute: type, owner, location, and reading value(s). Note, unlike the general Internet, searching IoT sensor data is not possible without meta-data. This is because Internet content searches operate on documents “encoded” in natural languages with inherent semantics—i.e., words whose meaning is generally agreed upon and there are dictionaries for interpretation. Sensor readings, on the other hand, are just numbers that depend on meta-data to provide context and semantics.

On a path to achieving interoperability, it is useful to agree on some common design guidelines and principles that facilitate interoperability, at the very least, by enabling translation between different systems. In our experience, a good start is to use named key, value pairs expressed in human readable form, such as JSON; agree on key categories of sensor and meta-data with a structured way to extend them; and use common naming of keys, at least within specific domains.

One way to arrive at the minimalistic set of interoperable items that need to be defined, is to articulate what needs to be provided to applications and services that are accessing an aggregate pool of multi-domain data at the API level. In addition to sensor data readings, apps would need to know things such as a unique sensor identifier, what type of information it reports, where it is or was located at the time of data capture, to whom it belongs, what are the privacy and security limitations for its use, etc. Applications need to be able to query sensor nodes and/or the cloud to obtain real-time readings/observations, and to query historical sensor values, say by name and time period. The system also needs to be bidirectional, so authorized entities can carry out actuation actions—such as opening or closing of a valve—for direct impact on the physical world as and when appropriate. For aggregations and collections

of data across devices and domains, it is highly desirable to be able to support searches of sensor data and meta-data by combinations of attributes (meta-data), such as sensors in my proximity, in an area (e.g., temperature sensors in the SW section on the third floor of this building), by location, by sensor type, by values/patterns/trends, by group/domain, etc.

A common and convenient form for retrieving, and possibly recording, such information is in the form of key, value pairs, such as:

```
<"sensor GUID": "xx">, <"reading": "yy">, <"units": "uu">  
<"time": "zz">, <"location": "g,e,o">, <"meta1": "abc">,  
<"meta2": "def">,...
```

Popular encodings like JSON have proven to be useful for this purpose. When communication bandwidth or storage capacity are at premium, binary encodings may be used internally. Getting some agreement on the naming of fields goes a long way toward interoperability and machine parsing that enables application portability. A good example of a domain-specific naming effort is the open-source [Haystack](#) project for smart buildings.

One of the key insights about interoperability is due to different design decisions in individual systems; it is almost impossible to achieve commonality at the data-structuring level, such as object definitions. Smart objects are a powerful abstraction within a given system, but interoperability is much easier by communicating individual attributes and values—e.g., by means of key-value mechanism—and then letting the receiving system map those to its own internal object representation.

The list of items and meta-data of interest can quickly grow to include: sensor type, location, frequency of reporting, mobile or static location, owner, domain, associations, access rights, privacy policy and restrictions, accuracy, manufacturer, model number, calibration, and others. The challenge is to devise a coordinated naming, taxonomy/ontology, and meta-data system that combine together to give minimal useful information in each observation, and allows the rest of the information of interest to be obtained by querying the sensor node or cloud data structures, as appropriate. Meta-data typically change at a different, and generally slower, rate than sensor data, which may favor separation of their processing and storage paths.

At the outset, cross-domain data and meta-data sharing is a challenging problem and it is tempting to stray into premature generalizations and formalisms that tend to be cumbersome due to incomplete understanding and may hinder adoption and implementation. Given the whole area is quite fluid and fast moving, probably the best and the fastest way forward is for

the IoT/WoT community to start by defining a minimal usable subset of guidelines and specifications with room for subsequent growth and expansion as our experience with building and operating those systems evolves, much like the evolution and success of the worldwide web. As Internet services have proved time and time again, the value of data increases with volume and diversity. The vision is to extend Internet technologies and experiences to sensors, and thus create universal sensor-enhanced world connectivity and realize the Internet-scale service promise and capability in IoT/WoT.

About the Author

Milan Milenkovic is a principal engineer and “intrapreneur” in Intel’s Internet of Things Group based in Silicon Valley. He has decades of experience at Intel and IBM working on a variety of complex systems in emerging technologies. He is also engaged with startup incubators and accelerators as a technical and business advisor through a Fulbright grant. Milenkovic received his M.Sc. degree in computer science from Georgia Institute of Technology and a Ph.D. in computer engineering from the University of Massachusetts. He is the author of a number of papers and books, and holds several U.S. and international patents. He is an ACM Distinguished Engineer and a senior member of IEEE.

DOI: 10.1145/2822643