Toward Interoperability in a Web of Things

Michael Blackstock
Media and Graphics
Interdisciplinary Centre (MAGIC)
University of British Columbia
FSC 3640 - 2424 Main Mall
Vancouver, BC, Canada V6T 1Z4
mblackst@magic.ubc.ca

Rodger Lea
Media and Graphics
Interdisciplinary Centre (MAGIC)
University of British Columbia
FSC 3640 - 2424 Main Mall
Vancouver, BC, Canada V6T 1Z4
rlea@magic.ubc.ca

UbiComp'13 Adjunct, Sept 8-12, 2013, Zurich, Switzerland.
Preprint

Abstract
In this position paper we explore the challenges and issues around interoperability in the web of things. A key concern is how to increase interoperability while maintaining a high degree of innovation and exploration in the community. To that end we propose a hub-centric approach toward interoperability consisting of four levels or stages. We are working to validate this approach in the context of a large-scale IoT ecosystem project consisting of eight IoT hubs in different domains where a key requirement is hub-to-hub and hub-application interoperability.

Author Keywords
Internet of Things; Web of Things; Interoperability

ACM Classification Keywords
C.2.6 Standards. H.3.5 Web-based services

General Terms
Standardization, Design

Introduction
Exposing things using the architecture of the web [5,7] promises to create a true ‘Internet of Things’. However, as the plethora of systems for the Web Of Things testifies, there is no standard approach to building and supporting WoT platforms today. While in a rapidly evolving area such as the WoT, premature standardization would risk killing innovation, equally, as

the community evolves it will need to seek some degree of interoperability if it is to offer developers more than simple islands of things leaving application developers to address interoperability issues themselves.

To that end, we have begun to explore WoT interoperability primarily with a goal of addressing three key questions:

- How can we increase interoperability in the Web of Things while maintaining innovation and rapid exploration by the WoT community?
- What are the different 'levels' of interoperability relevant to the WoT and what are their pros and cons for the community?
- Is there value in exploring an interoperability approach toward standardization and if so what interest, and appetite exists in the community?

We are developing our approach and validating it as part of the UK's Technology Strategy Board (TSB) funded 'Internet of Things Ecosystem Demonstrator' which has funded 8 consortia to each build an IoT hub and mandated they develop an agreed approach to IoT interoperability [12].

While we have no clear answers, we believe that as the community matures it needs to think about and address these questions. We offer our initial early experiences with the demonstrator project and leverage those experiences to propose a staged Hub centric approach to interoperability. Our goal with this position paper is to explore these questions and stimulate discussion and debate within the WoT community.

The Web of Things Today

The WoT approach leverages the unifying nature of the Web and so, to a certain extent, already offers some degree of interoperability and standardization. Things such as sensors and actuators can be represented as resources, and information about these things can be exchanged using the Representational State Transfer (REST) architecture of the web using the HTTP protocol.

It is equally clear that there has been an explosion in WoT platforms and systems over recent years. These efforts often take for granted that an interoperable IoT will make use of the web. While it is possible to give individual things or small groups of things a web presence, for example by providing a lightweight web server in an embedded device or a gateway, a growing trend today is to aggregate the web presence of many things using cloud-hosted platforms e.g. [3,9,13,14,15]. These larger scale 'WoT hubs' aggregate the web presence of many things for end users and application developers as illustrated in Figure 1. WoT hubs can be broken down into a number of categories which include *Web-enabled IoT products*, Web-centric *IoT development platforms*, *WoT Hubs* and *Sensor Webs*. Finally, *Mashup Platforms and Tools* may be used to combine hubs to create applications.

Web-enabled IoT products

A significant number of Internet-connected products today use web-based services to allow consumers to interact with them from mobile phone applications and web browsers e.g. [16]. The recently introduced 'Nest' thermostat [17], for example, is connected to a cloud service using home wifi networks permitting users to manage their home heating using their mobile phones and the web. Nest uses a web-based application programming interface (API) to access collected data and control its settings. Modern web-based APIs like the Nest's build on both the ubiquity of web protocols (HTTP) and the Representational State Transfer (REST) architectural style of the web [5].

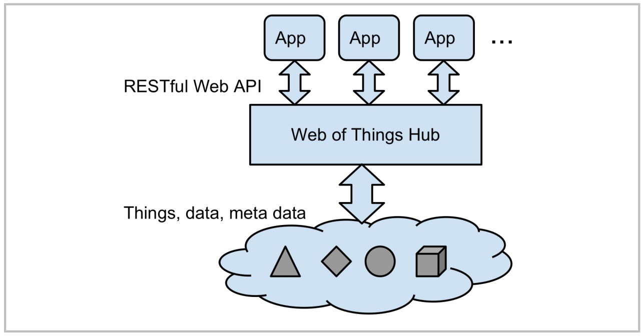

Figure 1. WoT Hubs provide a repository for one or more types of things.

Web-centric IoT platforms
Industrial IoT and M2M systems such as ThingWorx [18], Axeda [19], AirVantage [20] and RealTime.io [21] leverage web technologies for developers to create applications for the IoT. However, their focus is on providing a comprehensive set of tools for building end-to-end solutions. They generally use web technology to facilitate access, using a proprietary APIs and their own programming models.

WoT Hubs
Our own work, the WoTKit [3], as well as Xively, (formerly Cosm and Pachube [13]) aggregates collections of data streams called feeds to store information about sensors and the data they emit over time. Similarly, ThingSpeak [15] support a data model of channels similar to Xively and WoTKit feeds. All three include applications for processing, visualization and integration and offer the ability to find and share sensors and data, allowing others to take advantage of the integration work of others. The Open Sen.se platform acts as a hub for personal sensor data [14] and provides a set of embedded applications for users to track data generated from themselves and their things. Each of these platforms offer a 'hub' model to provide a repository for Things (data and metadata) and a set of APIs for accessing and using Things.

Sensor Webs
Sensor systems that heavily utilize web protocols have targeted large-scale sensor applications [4]. The Open Geospatial Consortium (OGC) has defined the Sensor Web Enablement (SWE) framework to provide a standard set of web service interfaces toward making it easier to share sensor data [10]. A new standard working group called Sensor Web Interface for the IoT [22] aims to link emerging web of things toolkits and platforms to the OGC SWE standards. There is a well-established need for coordinating systems to make it easier for applications to discover and access a wide variety of sensors independent of connectivity and data types.

Mashup Platforms and Tools
The use of web protocols by WoT hubs makes the creation of mashups applications possible. For example, by combining data from public weather sensors from one hub with data provided by the Nest thermostat 'hub', a developer can create an application that controls home heating and cooling based on the current and predicted weather patterns.

Mashup tools and platforms make it easier for developers to create applications by combining data and functionality across multiple physical and virtual data sources. While not all are exclusively targeted at IoT applications, several tools have emerged to ease the development of mashup applications. Yahoo Pipes [23] for example, is used to collect and process data using a dataflow-programming paradigm. The Mixup [11] tool from IBM integrates information from the web at the presentation layer. If This Then That (IFTTT) [24] is a platform for writing simple programs called *recipes* that react to changes in one web service (trigger) then do things in another (action). Paraimpu [25] aims to connect physical and virtual things to Web including arduino devices, social networks and other IoT platforms such as Xively. The WoTKit [3] includes a mashup tool called the Processor to combine and process data from sensor and social network feeds.

Some tools require integration work to collect data and control things in the real world, while others come with a variety of adapters 'out of the box'. While mashup tools can ease application development, they only provide a limited degree of interoperability between these web-connected 'islands of things'. Increasing interoperability can lead to increased use of internet-connected things benefitting device manufacturers, app developers and end users; however, there are several challenges that need to be addressed for a higher degree of interoperability to be achieved.

Interoperability Challenges

Given the different approaches taken by different WoT (and IoT) groups, it is clear that any developer wishing to cross system or platform boundaries needs to address and compensate for their different approaches. This requires they make assumptions about the types of things they want to connect to and compensate for different access and security methods, representations and URLs etc. While today this is perhaps not a significant issue, as the WoT evolves, we feel that this will increasingly hamper widespread adoption and lead to the inevitable 'islands of control' problem slowing innovation in the Web of Things.

To address this, it is worthwhile considering how to achieve some level of interoperability accepting that higher degrees of interoperability between hubs will require agreement on several fundamental issues such as the basic approaches to how 'things' are represented, found and accessed on the web. This will reduce the need for user involvement in configuration and setup, and specialized tools and adapters for application developers, making it easier to combine things from product specific 'hubs' that expose one or more catalogs of things to application developers.

Before addressing interoperability however, there must be some agreement on what interoperability means, the degree of interoperability required, and its implications for IoT system and application developers. What does it mean to be interoperable? Is the fact that an application developer can use a system to find, access and use things to create a mashup good enough? Perhaps we are already there! Once we've agreed on the answer (or answers) to this question, we can begin to resolve additional questions to move forward as a community.

What is a Thing? Today it's not clear what is really meant by 'things'. Are things limited to smart objects like sensors and actuators? Do they include anything with an electronic tag, or anything with an online presence like a street or bike rack represented by a row in an online databases? Do 'things' include people, and places [1,8]?

Thing Interaction. It is not clear how we should be able to interact with things. Do we only need the ability to interact with individual things, or query and control large groups of cooperating things at the same time?

Thing Identification. Once we've decided on what a 'thing' is, what is the minimum information necessary to identify these things? What are the appropriate resource identifiers, and representations for collections of things, things themselves, their properties and relationships?

Describing Things. How do we describe things? Do we need to establish a shared schema or ontology for things for greater interoperability, or can we find appropriate ways to involve users in connecting things and resolving ambiguities based on their current situation or context?

Finding Things. How do we find things? Should we be able to search for things by location, with certain properties? How can we discover and track mobile things that may move from one hub to another? Should we discover and access things through our friends using social networks [2,6]? How should things be organized?

Securing Things. How should we secure the access to non-public things? Can we make use of commonly used API security protocols like OAuth2 or are their other more appropriate security and authentication schemes? What are the various user and organizational roles necessary to control access to things?

Thing History. Should a 'hub' maintain the current state of a thing, or its past states? How far in the past? How is the current state of a thing related to the (meta) data they are associated with and generate.

To address these issues we discuss efforts approaches toward addressing interoperability in the IoT.

Existing Interoperability Approaches

Large IoT hub and platform vendors provide a degree of interoperability by providing a thing-agnostic model and API to integrate things across a wide variety of domains. These vendors aim to create a network effect where the hub becomes more valuable as more users and their things are connected. By doing so vendors hope to establish their hubs as de-facto standards for web of things interoperability.

The Internet of Things Architecture project (IoT-A) is proposing an architectural reference model for IoT interoperability together with key components of the future IoT to enable search, discovery, and interaction as one coherent network [26]

The Open Geospatial Consortium also has ongoing efforts to increase interoperability in the IoT. The Sensor Web for IoT working group has begun work toward integrating their location and Sensor Web Enablement standards with the IoT [10,22]

Domain-specific sensor data portals have established standards toward interoperability. For example, the International Federation of Digital Seismograph networks (FDSN) publishes web service specifications and XML schema to provide a degree of interoperability across seismic sensor data portals to make this data easier to consume by applications [27].

The IETF community has been involved in foundational IoT technologies such as IPv6 and the Constrained Application Protocol (CoAP) focusing on getting constrained devices and sensor networks connected to the Internet. Similarly, the IEEE has several protocol standards that form the foundation of the IoT providing connectivity between things and the Internet [28].

While these efforts are moving the IoT toward greater interoperability, some, such as the IETF and IEEE deal primarily with connecting things to the Internet and the web specifying only the core networking infrastructure and protocols needed, not hub-to-hub or hub-to-application interoperability. In an attempt to address both current and future needs of the IoT, ambitious efforts such as the IoT-A project could move forward too quickly introducing complex requirements without establishing a clear need. Vendors may favour less complex approaches that address requirements as they emerge. If we leave large proprietary hub vendors to establish standards, we risk innovation from smaller companies, often the primary source of innovation in emerging markets.

We propose that the community move forward with a lightweight approach consisting of four different levels or stages of interoperability. In the next section we describe this path to interoperability and our ongoing work to validate this approach in cooperation with eight domain-specific hubs in the UK rooted in practical experience.

A Path to Interoperability

Given the number of challenges, and that the WoT is still in its early stages of development, is clear that all of the outlined challenges cannot be addressed without significant cooperation and widespread agreement on common models, approaches and best practices. To move forward it will be important to understand the benefits and costs of greater interoperability such as cost savings and possible added complexity and restriction on innovation. A methodology that can capitalize on interoperability while maintaining a high degree of innovation is needed.

Our position is that the best basis for such a methodology is to adopt a hub-centric model for the WoT that uses mashup as the fundamental driver for hub interoperability. We believe that such an approach will address the current needs of the majority of the WoT community, who are currently focused on developing applications against single platforms and who only require interoperability to allow them to easily reach out to other WoT systems and access data on specific things that their core applications need.

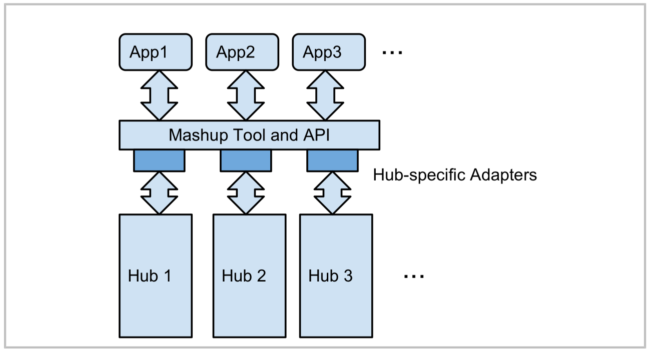

Figure 2. Mashup Platforms used to address WoT hub interoperability.

To leverage the Hub and Mashup model, we suggest a staged approach which initially agrees on foundational technologies and approaches, then later builds on these as the community evolves, moving from to more comprehensive levels of interoperability, reducing the need for hub-specific adapters. Of course, some groups and domain-specific IoT communities can (and have) moved forward more quickly, but without widespread adoption, these communities may be forced to move back or build adapters when attempting to integrate with hubs from other IoT domains.

We suggest that there are four stages or levels toward WoT hub interoperability:

Web of Things Core. At this initial stage, hubs expose things and catalogs of things using the web architecture allowing application developers to create applications using RESTful web services. This will require integration by application developers, or using adapters with mashup tools as shown in Figure 2. This level offers minimal interoperability and allows WoT application developers to find IoT hubs but requires the application developer or the mashup platform to do the 'heavy lifting' in understanding how to use these Hubs and the things they manage.

Web of Things Model. The next stage of interoperability requires agreement on basic approaches and models, assuming a user (i.e. an application developer or end user) is still in the loop. At this level, there is agreement on a basic set of mechanisms to determine what things and data a hub manages, but no agreement on the details of those things. In essence, this stage provides a basic catalogue model allowing application developers to use a common API to retrieve a high level catalogue of the things a hub contains. This stage will facilitate the development of generic adapters and the required tools for integration and interoperability between hubs will become simpler.

Web of Things Hub. The third stage of development requires agreement on some implementation issues such as concrete representations, URLs, standard schemes for describing and getting data from things in a generic fashion. It also requires a basic agreement on security mechanisms so that Hubs can control access to things and offer some guarantees over who is accessing, providing and using things and their data. Reaching this stage of interoperability allows application can do more automatically with less intervention.

Web of Things Profiles. This final stage of interoperability model is reaching agreement on detailed thing profiles. This level requires agreement on the complex semantics of things, e.g. agreeing that a temperature sensor in one hub is the same as one in another hub, and agreeing on the types of data each manages. Essentially the taxonomy of things and the ontological models that hubs support will need to be defined. By reaching agreement at this level, deep integration of application is possible allowing hubs and things to link to and communicate directly with each other as illustrated in Figure 3.

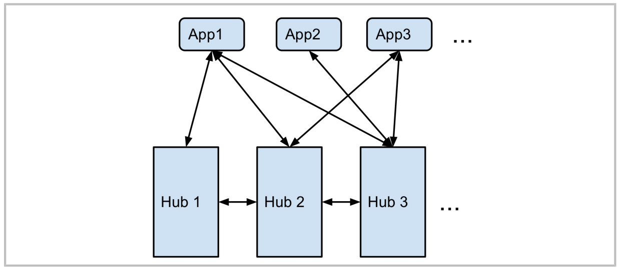

Figure 3. Complete hub-interoperability where applications can access hubs directly. Hubs can exchange information as needed to find and interact with things across hubs.

Validating Web of Things Interoperability

As part of an exploration of these levels of interoperability, we are working with a group of eight IoT hubs, each being built by a consortium of industry and academic partners in the UK. Each Hub consortium is addressing a different domain or geographical area but each has agreed to work towards a level of interoperability with the other Hubs. These eight hubs are part of the *Internet of Things Ecosystem Demonstrator* project funded by the UK Technology Strategy Board (TSB). Which has established a large scale, open cross-domain IoT program where one of the primary goals is interoperability between open IoT information hubs to "stimulate the development of an open application and services ecosystem in the Internet of Things (IoT)" [12].

Each hub in the project is industry led, and consists of a mix of large and small companies supported by one of more academic partners. Areas of focus for the clusters include Transportation, the Built Environment, Embedded devices, the Home, Critical infrastructure, and Education. Further details can be found at https://connect.innovateuk.org/web/internet-of-things-ecosystem-demonstrator.

Our focus, as part of the *Smart Streets* consortium, is looking at critical infrastructure with a focus on road networks. To that end we are developing an IoT hub that manages things and data ranging from road pavement conditions, lights and signals to drains and flooding patterns. For these things the hub is managing both real time and non-real time sensors and assets status and updates from across the UK's road network.

Within the framework of the TSB project, members have established agreement on the use of web technologies to expose the things available on their hub and are, in many cases, using existing WoT platforms. The Smart Streets project for example is making use of the WoTKit [3] as its WoT platform augmented by an open source data hub called CKAN [29] to host static data needed by developers. Other hubs such as STRIDE, looking at Container ports and shipping information use the BT hub. The DISTANCE project is using the Xively (Cosm) hub to expose things and data to educational establishments. All are leveraging the web to make their data available to application developers and other hubs. Clearly, these hubs are at or past the first stage of interoperability - "WoT Core".

As part of the IoT ecosystem demonstrator project each hub is participating in an interoperability working group to establish the basic requirements and models for inter-hub interoperability. The goal is to specify the minimum requirements needed for hubs to exchange data and application developers to access and use data from each hub in a consistent manner. This involves understanding what is common about each hub, how they expose things, and how application developers will use the hubs individually and together. An early deliverable will outline the basic requirements and model shared by IoT hubs toward interoperability – "WoT Model".

From that Interoperability working group, early work has begun by agreeing on specific implementation issues such as defining how things and catalogs of things are represented and searched for in a hub. The aim is to define simple representations that can be easily mapped to existing implementations using a facade and adapters so that hubs can meet interoperability requirements and its associated benefits with little effort and minimal complexity, moving to the "WoT Hub" stage.

Because of the variety of WoT domains, it seems unlikely that hub-hub interoperability can be increased further in the lifetime of this initial project. Once basic application and hub-to-hub communication is established, we are optimistic that a follow on project could begin to explore deeper semantic interoperability between hubs.

Conclusions

Today companies and organizations are hosting web of things hubs that aggregate catalogs of things and provide web-centric APIs for application developers. Clearly, aggregating things and their associated communities around hubs makes application development easier. It is also clear that degrees of hub-interoperability are possible using mashup tools and applications. The question is: How do we move forward as a community from simply leveraging the web toward stronger interoperability between WoT hubs?

We proposed four levels or stages along a path. This path allows application developers to create mashup applications using WoT hubs today while we work toward greater interoperability between hubs over time. In the context of the IoT Ecosystem Demonstrator project, we hope to gain a better understanding of how to create an interoperable web of things, and how (and when) the web of things may evolve from simply exposing things to the web toward a more standards-based approach while maintaining innovation in the WoT community.

Acknowledgements

We would like to thank our colleagues in the IoT Ecosystem demonstrator project whose discussions have helped us form our position reported here. Partial support for this work has come from NSERC Canada.

References

[1] Blackstock, M., Kaviani, N., Lea, R., and Friday, A. MAGIC Broker 2: An open and extensible platform for the Internet of Things. Internet of Things (IOT), 2010.

[2] Blackstock, M., Lea, R., and Friday, A. Uniting online social networks with places and things. Workshop on the Web of Things (WoT 2011).

[3] Blackstock, M. and Lea, R. IoT Mashups with the WoTKit. Internet of Things 2012. (2012).

[4] Botts, M. and Robbin, A. Bringing the Sensor Web together. Géosciences 2007, 6 (2007), 46–53.

[5] Fielding, R.T. and Taylor, R.N. Principled Design of the Modern Web Architecture. ACM Transactions on Internet Technology (TOIT) 2, 2 (2002), 115–150.

[6] Guinard, D., Fischer, M., and Trifa, V. Sharing using social networks in a composable Web of Things. IEEE Web of Things Workshop at PerCom 2010.

[7] Guinard, D., Trifa, V., et al. From the Internet of Things to the Web of Things: Resource Oriented Architecture and Best Practices. In Architecting the Internet of Things. Springer, 2011, 97–129.

[8] Kindberg, T., Barton, J., Morgan, J., et al. People, places, things: Web presence for the real world. IEEE Mobile Computing Systems and Applications 2000.

[9] Liqian, L., Kansal, A., Nath, S., and Zhao, F. SenseWeb: Sharing and Browsing Environmental Changes in Real Time. http://research.microsoft.com/apps/pubs/default.aspx?id=77850. Retrieved May 30, 2013

[10] Reed, C.; Botts, M.; Davidson, J.; Percivall, G.;, "OGC® Sensor Web Enablement:Overview and High Level Architecture.," Autotestcon, 2007 IEEE, pp.372-380, 17-20 Sept. 2007

[11] Yu, J., Benatallah, B., et al. Mixup: A Development and Runtime Environment for Integration at the Presentation Layer. Web Engineering 2007, Springer-Verlag (2007), pp. 479–484.

[12] Internet of Things Ecosystem Demonstrator. https://connect.innovateuk.org/web/internet-of-things-ecosystem-demonstrator. Retrieved May 30, 2013

[13] Xively – Public Cloud for the Internet of Things. https://xively.com/. Retrieved May 30, 2013

[14] Open Sen.se Feel, Act, Make sense. Feel, Act, Make sense. http://open.sen.se/. Retrieved May 30, 2013

[15] The Internet of Things - ThingSpeak. https://thingspeak.com/. Retrieved May 30, 2013

[16] Internet of Things and Kickstarter. A perfect match?. http://postscapes.com/internet-of-things-and-kickstarter. Retrieved May 30, 2013

[17] Nest - The Learning Thermostat. http://nest.com. Retrieved May 30, 2013

[18] ThingWorx – The 1st Application Platform for the Connected World. http://www.thingworx.com/. Retrieved May 30, 2013

[19] Axeda Application and Data Integration Platform. Axeda Application and Data Integration Platform. http://www.axeda.com/. Retrieved May 30, 2013

[20] AirVantage M2M Cloud Platform. http://www.sierrawireless.com/productsandservices/AirVantage.aspx. Retrieved May 30, 2013

[21] The Real-Time Internet of Things Platform - RealTime.io. http://realtime.io/. Retrieved May 30, 2013

[22] Sensor Web Interface for IoT SWG | OGC(R). http://www.opengeospatial.org/projects/groups/sweiotswg. Retrieved May 30, 2013

[23] Pipes: Rewire the Web. http://pipes.yahoo.com/pipes/. Retrieved May 30, 2013

[24] IFTTT - Put the internet to work for you. https://ifttt.com/. Retrieved May 30, 2013

[25] Paraimpu - The Web of Things is more than Things in the Web: Paraimpu.crs4.it. http://paraimpu.crs4.it/. Retrieved May 30, 2013

[26] Internet of Things - Architecture — IOT-A. http://www.iot-a.eu/public. Retrieved May 30, 2013

[27] FDSN Publications. http://www.fdsn.org/publications.htm. Retrieved May 30, 2013

[28] IEEE-SA - Internet of Things Related Standards. http://standards.ieee.org/innovate/iot/stds.html. Retrieved May 30, 2013

[29] ckan - The open source data portal software. http://ckan.org/. Retrieved May 30, 2013