Leveraging the Web Platform for the Web of Things: Position Paper for W3C Workshop on the Web of Things

Erik Wilde (dret@berkeley.edu),
Florian Michahelles (florian.michahelles@siemens.com), and
Stefan Lüder (stefan.lueder@siemens.com)

April 2014

Abstract

Web Architecture provides a general-purpose way of establishing an interlinked network of resources, which are interacted with through the exchange of representations of their state. We argue that the "Web of Things" fits well into this general framework, and thus should be built firmly on the foundation provided by Web Architecture. We also argue that in order to allow an evolutionary path towards a "Web of Things", it is important to take small and incremental steps towards the final goal, instead of trying to establish a grand "Web of Things Architecture" in one monolithic step. One interesting first step could be to focus on Activity Streams as one way how streams of resource updates can be represented in a uniform, extensible, and machine-readable way.

Contents

1 Our Perspective

For the "Web of Things" to become a global reality, there needs to be an open, modular, extensible, and flexible set of technologies that are in line with the *Architecture of the World Wide Web* [10], and can be used as a platform to build on. As a result, this platform should be a RESTful SOA, so that it can be easily extended, and decentralized applications can be built on top of it.

For a while, it looked as if *Atom* [16] might grow into a good foundation, and there was an active community and a dynamic and rapidly evolving landscape of Atom-related technologies.[1] However, the ascent of JSON and the fact that all Atom-related technologies were XML-based meant that regardless of its utility, Atom's popularity declined. By now it seems that feeds are on the decline in general, both as machine-readable services provided by Web sites, as well as a foundation for SOA designs.

Given the realities of Atom's decline, it seems that something JSON-based might replace Atom, so that the same kind of capabilities can be exposed, but based on a more popular metamodel. We believe that *Activity Streams* may grow into such a foundation. Some submissions for W3C's 2013 *Workshop on Social Standards*[2] specifically mentioned Activity Streams and generally the fact the social standards should support M2M scenarios, and provide a solid foundation for BI platforms and other SOA scenarios.

Building on a number of years of experience with organizing the successful *Web of Things (WoT) workshop series* [6, 7, 13, 14] (held in conjunction with a variety of conferences), it seems that there is a lot of interest and a substantial community that is committed to using the Web platform for new and challenging scenarios. Specifically, there seems to be a difference in perspective between the *Internet of Things (IoT)* community, and the *Web of Things (WoT)* community, even though both terms probably by now have become buzzwords and as such are increasingly used as marketing rather than technical terms. But still, we claim that there is a difference, and that this difference matters:

- *Internet of Things (IoT):* The main goal is to establish connectivity, often taking into account the specific constraints of embedded systems, such as limited computing power or connectivity. Typical examples of activities in this space are 6LoWPAN [11] or CoAP [19]. The focus of typical IoT activities is on establishing connectivity at a certain protocol level.
- *Web of Things (WoT):* The main goal is to use Web Architecture to interlink resources such as sensors and actuators with other Web resources. Connectivity is out of scope: whether a sensor is directly connected via IoT, or instead is available through a gateway with some proprietary connection behind it does not matter, as long as it has a URI, a representation, and exposes its relations and interactions via hypermedia. The focus on typical WoT activities is on establishing a platform for building Web-level applications.

Our perspective is that it would be a timely and useful endeavor to take typical scenarios from the Web of Things, and "test" the current landscape of available technologies against them. Which problems can be solved based on existing technologies, which solutions might benefit from well-defined design patterns and best practices, and in which cases may new technologies be required to get to truly open solutions? Possible scenarios might include (but should be limited to):

- *Environmental Scenarios:* In scenarios from agriculture to water management, it becomes increasingly clear that resources can be utilized more responsibly when assisted by advanced planning, monitoring, and resource allocation methods.
- *Building Automation:* While the computerization of commercial buildings is a reality, it currently is rather closed, and not easily connected across building boundaries, or with private homes. Energy consumption of buildings can be better optimized the smarter they get, and that also means the more openly connected they get.
- *Healthcare Settings:* Hospital IT is a reality today, but like building automation mostly based on closed architectures. Increasingly, information should be shared by local practitioners as well as by patients themselves (using wearables to collect data), and this more heterogenous landscape of services and devices will need new architectural approaches.

[1] http://dret.typepad.com/dretblog/atom-landscape.html
[2] http://www.w3.org/2013/socialweb/

- *Urban Environments:* Urban areas are densely populated with people as well as with services and devices, providing information from sources as diverse as restaurants, public transport, traffic sensors, and air/water quality. The more easily this information can be used and recombined, the easier it will be to improve the quality of life in cities by helping people to make smarter decisions.

The more all of these scenarios can be built on the same foundations, the easier it will be for those visions to become reality, and for innovation to cut across those scenarios and start combining things and services in ways that would have been impossible to predict.

Siemens as one of the major global players in many of the above scenarios has a substantial interest in understanding long-term trends and customer needs and requirements. An open design facilitates both a more loosely coupled and agile internal development, and satisfies the openness and flexibility of solutions that customers increasingly demand.

In a variety of "Web of Things" scenarios (such as industry automation, smart buildings, smart grid, healthcare, and transportation), an important goal is to both participate in and contribute to the ongoing evolution of the Web as a connective fabric that connects more and more aspects of modern life.

2 Our Interests

For a major player in a variety of IT-intensive scenarios, it is important to participate in and contribute to the advancement of these fields. In sectors such as building automation and healthcare IT it is clear that the landscape is changing. What used to be complex but relatively closed and controlled systems, increasingly need to be ecosystems designed for openness, flexibility, and agility.

Another relevant aspect is that Web Architecture facilitates openness, but does not require it. This means that if for business reasons it is decided that some system should not provide complete openness, Web Architecture will still provide many advantages as a way to design and build systems. It promotes ecosystems that make it easier to evolve and extend existing systems, and thus also provides an excellent foundation for internal architecture.

Our goal is to help develop the vision and a path for the "Web of Things". We believe that this can be partly done with promoting best practices and design patterns, but that in some cases there also may be a need for additional standards.

For example, streaming of continuously updated data is a common use case in "Web of Things" scenarios. Currently, a variety of protocols better suited than the request/response model of HTTP/1.1 [3] can be used to support such as scenario, for example protocols such as *MQ Telemetry Transport (MQTT)* [12], *Extensible Messaging and Presence Protocol (XMPP)* [18], *WebSocket* [2], *HTTP/2* [1], *PubSubHubbub* [5], or *Server-Sent Events* [8]. Furthermore, in addition to those streaming protocols, it may be interesting to have standard subscription mechanisms, and maybe even filtering, so that clients have a way to subscribe to their preferred subset of resource state changes. It would be good for those scenarios to be described in detail, so that implementers can either simply follow prescribed design patterns, or maybe additional specifications can help to increase interoperability.

We hope that this workshop will result in the formation of an activity and/or a working group, and we expect this working group to have a critical impact on the evolution and adoption of the "Web of Things".

2.1 Challenges

One of the challenges for a "Web of Things" activity will be to decide which issues to tackle in which sequence. Given the overall constraints of Web Architecture and *Representational State Transfer (REST)* [4] as the underlying architectural style, many problems could be tackled well within the boundaries of existing standards. Designing new media types in many cases may already cover quite a bit of ground.

However, it may be the case that certain areas are rife for consensus, and should be tackled first. One possible example is the case of *Activity Streams* mentioned above. Atom failed after a promising start because it was built on a foundation that saw a heavy decline in popularity (XML). With the gap left by Atom, it seems that both the lessons learned, and the scenarios, may warrant to give the idea another chance, this time on a more modern foundation.

Apart from designing the general interaction fabric, another challenge might be to resist the urge to create too many and too specific models. Learning from the Web itself may be an interesting lesson here: There is one very generic model that has no domain-specific facilities other than basic relations and interaction patterns. Everything else then can develop within this generic framework. It may be the case that the same pattern should be applied here: provide very generic and extensible models that provide support for basic interaction patterns, and then see how these are applied in real-world scenarios.

Another challenge might be the lack of registries. In the IETF, a common and successful specification design pattern has been to decouple protocols and protocol field values. IANA runs many registries, many of them on network layers underneath the Web, but over the past several years, an increasing number of Web-level standards have established their own registries.[3] Depending on the kind of developments happening in the "Web of Things" space, it might make sense to create a framework for establishing, maintaining, and publishing registries at the W3C.

2.2 Risks

The biggest risk in this space might be to try to solve too many problems at the same time, and/or with the same specification. The Internet and the Web thrive because they are ecosystems, where certain foundations can grow and develop because they are useful and popular, while others will remain unused and disappear if there proves to be little demand. Following this pattern of small and independent components will help the "Web of Things" to evolve in the same way as the Web itself, where apart from the very basic specifications for URIs, HTTP, and HTML, pretty much every other component has seen some competition over the years.

Using the mechanisms of loose coupling [17] and media types, this risk can be mitigating by allowing runtime negotiation. Using the mechanism of registries, some basic agreements can be hard-coded into a specification, while there still is room to add more consensus-based agreements later on. The main risk here is to create something that is too monolithic and static, and in particular in a space with as much momentum as the "Web of Things", this could result in an architecture that would not be able to keep up with the rapidly changing field.

One additional risk is the development of domain vocabularies. W3C should probably not get into the business of developing domain vocabularies. Instead, it should focus on the framework within which domain experts then can develop and use their vocabularies. It is then up to communities interested in certain domains to either create standards or informal documents which allow them to use the vocabulary that they need for their scenarios.

2.3 Standards

As outlined in the previous two sections, the main challenge and associated risks will be to make sure that only core aspects are tackled in standards work. For example, W3C's nascent *Social Web Working Group*[4] will probably have the development of a stable *Activity Streams* [20][5] specification as one of its core deliverables, and possible work in the "Web of Things" space should probably follow their work, and make contributions (or define extensions) where necessary. Activity Streams and related specification such as *Action Handlers* [21] might be the interaction fabric that will be one of the enablers for a global "Web of Things".

In the space of standards work, it might also be beneficial to analyze and describe related work, such as the *OPC Unified Architecture (OPC UA)* [9]. In many cases, these architectures might have some overlap with a more open and modular "Web of Things" architecture. It would help standardization work a lot to make these overlaps explicit, so that specific communities that already use domain-specific standards have an easier way to understand how they relate.

[3] One popular example is *Web Linking* [15] and its associated registry of well-known link relation values at http://www.iana.org/assignments/link-relations/link-relations.xhtml.

[4] http://www.w3.org/2013/socialweb/social-wg-charter.html

[5] While the last currently available public draft is an IETF track document, the upcoming Social Web Working Group will result in this specification getting onto the W3C track.

2.4 Issues

We believe that the main issue to be tackled first is the interaction fabric. This may be based on Activity Streams, but it also may need additional features such as discovery, configuration, subscription management, push models, filtering languages, and aggregation models. If and how all of these issues are tackled in the context of a possible working group remains to be seen, and will mostly depend on interested parties, and the scenarios that they contribute.

2.5 Possible Solutions

Like the Web itself, the "Web of Things" is not something that is invented or created; it is a vision of a Web that's a friendlier place for things to be in. For this vision to become a reality, we believe that it is necessary to focus on evolving the existing Web in the same evolutionary and loosely coupled way as the Web has evolved so far. There is quite a bit to learn from the existing Atom landscape, which probably should be used as a valuable starting point of a collection of solutions that were created in response to real problems. Activity Streams may follow a similar evolutionary path.

In addition to this basic interaction fabric, it may be time to approach the "push problem" more aggressively than so far. It is possible to approach this problem in ways that are in line with REST and Web Architecture, but so far there has not been a single answer that developers can safely use. There are a variety of potential candidate protocols, and there also is an existing collection of proprietary push services for mobile platforms.[6] It may be time for the Web platform to include an answer to this problem, so that developers as well as tools can develop applications involving push interactions more robustly.

It may not even be necessary for the W3C to step in and define or refine one of the existing protocols (as briefly discussed in Section 2). But it may be interesting to look at ways how resources can advertise their push capabilities, so that there is a robust way how push capabilities can be detected, and how, if there is a match in protocol support, clients can subscribe to such a push mechanisms.

3 Possible Work Items

As outlined in the previous section, depending on scenarios and interests, there are a variety of possible work items in the "Web of Things" space. One of the unique strengths of the W3C's workshop model is that their high visibility allows to gauge the interest of the global community. Based on the participation and position papers at the workshop, it will be possible to create a roadmap for possible work items, based on the scenarios of the participants, and the gaps they perceive in terms of guidance or standards.

4 Conclusions

It seems that 2014 is becoming the year of the "Internet of Things" and the "Web of Things". As these terms enter mainstream media, they also become buzzwords and their meaning becomes increasingly fuzzy. However, the building momentum makes this a very good moment in time to collect community feedback about perceived gaps and shortcomings of the current Web landscape. It seems that the "Web of Things" momentum may even warrant the creating of a new activity, which could then coordinate various working groups focusing on specific issues such as the general interaction model, or how to use push on the Web. Whatever the result of the workshop will be, we are sure that now is a good time to address the "Web of Things" topic at the W3C, and it is very likely that whatever the result will be, it will extend the Web to be more inclusive than ever before.

References

[1] Mike Belshe, Roberto Peon, and Martin Thomson. Hypertext Transfer Protocol Version 2. Internet Draft draft-ietf-httpbis-http2, April 2014.

[6] Examples for these services include iOS's *Apple Push Notifications (APN)* and Android's *Google Cloud Messaging (GCM)*.

[2] Ian Fette and Alexey Melnikov. The WebSocket Protocol. Internet RFC 6455, December 2011.

[3] Roy Thomas Fielding, Jim Gettys, Jeffrey C. Mogul, Henrik Frystyk Nielsen, Larry Masinter, Paul J. Leach, and Tim Berners-Lee. Hypertext Transfer Protocol — HTTP/1.1. Internet RFC 2616, June 1999.

[4] Roy Thomas Fielding and Richard N. Taylor. Principled Design of the Modern Web Architecture. *ACM Transactions on Internet Technology*, 2(2):115–150, May 2002.

[5] Brad Fitzpatrick, Brett Slatkin, Martin Atkins, and Julien Genestoux. PubSubHubbub Core 0.4, June 2013.

[6] Dominique Guinard, Vlad Trifa, and Erik Wilde, editors. *Second International Workshop on the Web of Things (WoT 2011)*, San Francisco, California, June 2011.

[7] Dominique Guinard, Erik Wilde, and Vlad Trifa, editors. *First International Workshop on the Web of Things (WoT 2010)*, Mannheim, Germany, March 2010.

[8] Ian Hickson. Server-Sent Events. World Wide Web Consortium, Candidate Recommendation CR-eventsource-20121211, December 2012.

[9] International Electrotechnical Commission. OPC Unified Architecture — Part 1: Overview and Concepts. IEC 62541-1, February 2010.

[10] Ian Jacobs and Norman Walsh. Architecture of the World Wide Web, Volume One. World Wide Web Consortium, Recommendation REC-webarch-20041215, December 2004.

[11] Nandakishore Kushalnagar, Gabriel Montenegro, and Christian Peter Pii Schumacher. IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals. Internet RFC 4919, August 2007.

[12] Dave Locke. MQ Telemetry Transport (MQTT) V3.1 Protocol Specification. Technical report, IBM, August 2010.

[13] Simon Mayer, Dominique Guinard, and Erik Wilde, editors. *Third International Workshop on the Web of Things (WoT 2012)*, Newcastle, UK, June 2012.

[14] Simon Mayer, Vlad Trifa, Dave Raggett, and Dominique Guinard, editors. *Fourth International Workshop on the Web of Things (WoT 2013)*, Zürich, Switzerland, September 2013.

[15] Mark Nottingham. Web Linking. Internet RFC 5988, October 2010.

[16] Mark Nottingham and Robert Sayre. The Atom Syndication Format. Internet RFC 4287, December 2005.

[17] Cesare Pautasso and Erik Wilde. Why is the Web Loosely Coupled? A Multi-Faceted Metric for Service Design. In Juan Quemada, Gonzalo León, Yoëlle S. Maarek, and Wolfgang Nejdl, editors, *18th International World Wide Web Conference*, pages 911–920, Madrid, Spain, April 2009. ACM Press.

[18] Peter Saint-Andre. Extensible Messaging and Presence Protocol (XMPP): Core. Internet RFC 6120, March 2011.

[19] Zach Shelby, Klaus Hartke, and Carsten Bormann. Constrained Application Protocol (CoAP). Internet Draft draft-ietf-core-coap-18, June 2013.

[20] James M. Snell. JSON Activity Streams 2.0. Internet Draft draft-snell-activitystreams-05, November 2013.

[21] James M. Snell and Matthew Marum. JSON Activity Streams 2.0 — Action Handlers. Internet Draft draft-snell-activitystreams-actions-03, July 2013.