Network Working Group
Internet-Draft
Intended status: Informational
Expires: May 18, 2017

J. Jimenez
Ericsson
H. Tschofenig
ARM
D. Thaler
Microsoft

November 14, 2016

Report from the Internet of Things (IoT) Semantic Interoperability (IOTSI) Workshop 2016
draft-iab-iotsi-workshop-01.txt

Abstract

This document provides a summary of the 'Workshop on Internet of Things (IoT) Semantic Interoperability (IOTSI)' [IOTSIAG], [IOTSIWS], which took place in Santa Clara, CA, on March 17-18, 2016. The main goal of the workshop was to foster a discussion on the different approaches used by companies and standards developing organizations to accomplish interoperability at the application layer. This report summarizes the discussions and lists recommendations to the standards community. The views and positions in this report are those of the workshop participants and do not necessarily reflect those of the authors and the Internet Architecture Board (IAB), which organized the workshop.

Status of This Memo

Table of Contents

1. Introduction

The Internet Architecture Board (IAB) holds occasional workshops designed to consider long-term issues and strategies for the Internet, and to suggest future directions for the Internet architecture. The investigated topics often require coordinated efforts of many organizations and industry bodies to improve an identified problem. One of the targets of the workshops is to establish communication between relevant organizations, specially when the topics are out of the scope for the Internet Engineering Task Force (IETF). This long-term planning function of the IAB is complementary to the ongoing engineering efforts performed by working groups of the IETF.

Increasing interoperability in the area of Internet of Things (IoT) has been a top priority for many standards organizations and pariticularly the lower layers of the Internet protocol stack have received a lot of attention. Also at the application layer, such as

with CoAP and HTTP, there is a trend in reusing RESTful design patterns. However, data exchanged on top of these application layer protocols there is still a lot of fragmentation and the same degree of increase in interoperability has not been observed.

Thus, the IAB decided to organize a workshop to reach out to relevant stakeholders to explore the state-of-the-art and to identify communality and gaps.

- o The different perceived state of the art on data and information models.

- o The lack of an encoding-independent standardization of the information, the so-called information model.

- o The strong relationship with the underlying communication pattern, such as remote procedure calls (RPC), publish/subscribe or RESTful designs.

- o Identifying which similar concepts groups have develop in parallel, specially those that would require only slight modifications to solve interoperability.

- o Identifying how existing data models can be mapped against each other to offer inter working.

- o Identifying common use cases for cooperation and harmonization.

2. Terminology

The first roadblock to semantic interoperability is the lack of a common vocabulary to start the discussion. There is a need to align between participating organizations on a common set of basic terms. [RFC3444] does a start by separating conceptual models for designers or Information Models (IMs) and concrete detailed definitions for implementors or Data Models (DMs). There are concepts that are undefined in that RFC and elsewhere, such as the interaction with the resources of an endpoint or Interaction Model. Therefore the three "main" common models that could be identified were:

Information Model (IM) it defines an environment at the highest level of abstraction, they express the desired functionality. They can be defined informally (e.g. in plain English) or more formally (e.g. UML, Entity-Relationship Diagrams, etc.). Implementation details are hidden.

Data Model (DM) it defines concrete data representations at a lower level of abstraction, including implementation and protocol-

specific details. Some examples are: SNMP Management Information Base (MIBs), W3C Thing Description (TD) Things, YANG models, LWM2M Schemas, OCF Schemas and so on.

Interaction Model (IN) it defines how data is accessed and retrieved from the endpoints, being therefore tied to the specific communication pattern that the system has (e.g. REST methods, Publish/Subscribe operations or RPC-calls).

Another identified terminology issue is the semantic meaning overload that some terms have. Meaning will vary depending on the context the term is used. Some examples of such terms are: semantics, models, encoding, serialization format, media types or encoding types. Due to time constraints no concrete terminology was agreed upon, however work will continue within each organization to create various terminology documents, see [IOTSIGIT].

3. Architecture

Architectures follow different design patterns, some are REST-oriented with clear methods to find and manipulate resources on endpoints with almost no state kept between them. Others are more Publish/Subscribe-oriented that focus on the flow of information based on the topics of the information that is being shared. Others are RPC-like requiring to know beforehand the set of parameters that are accessed, requiring to generate code both on the client and server side, they are tightly coupled and service-oriented.

Thing-hub-cloud things talk to a hub, which talks back to them and to the cloud. There is a spectrum of emphasis on the hub vs. the cloud. For some the hub is simply an access point; for others it is a critical place for control loops and ALGs.

Meta Thing some things are actually virtual, like an alarm composed of clock, lights, and thermostat.

Nearby things some things may be geospatially related even if they are not on the same network or within the same administrative domain.

Current data models and definitions for "things" often focus on defining actual physical devices and representing their state. That focus should perhaps be shifted into a more holistic or user-oriented perspective, defining abstract concepts as well. On the other a lot of focus is placed on human interaction with physical devices while IoT will be more oriented to interaction between "things" instead.

Those things should be designed to be multiple-purpose, keeping in mind that solutions should be open-ended to facilitate new usages and new future domains of operation. this pattern has already happened, devices that were intended for one purpose end up being used for a different one given the right context ((e.g. smart phone led light turned out to be a heartrate monitor). IoT domain is currently missing insights into what user actually expect.

4. What Problems to Solve

Although the workshop focused on a specific problem it became obvious from the position papers that various organizations, industry groups and individuals attempted to solve different problems. At least the following goals have been described:

o Formal Languages for Documentation Purposes

Standardization organizations are in the need for a more formal description of their information and data models in order to simplify review, and publication. For example, the Open Mobile Alliance (OMA) used an XML schema [LWM2M-Schema] to describe their object definitions (i.e., data model) as XML instance documents. These XML documents offer an alternative way of describing objects compared to the tabular representation found in the specification itself. The XML files of standardized objects are available for download at [OMNA]. Furthermore, a tool is offered to define new objects and resources. The online editor tool can be found at [OMA-Editor].

o Formal Languages for Code Generation

Formal data and information modelling languages for use by developers to enable code generation. For example, the Allseen Visual Studio Plugin [Allseen-Plugin] offers a wizzard to generate code based on the formal description of the data model. Another example of a data modelling language that can be used for code generation is YANG. A popular tool to help with code generation of YANG modules is pyang [PYANG].

o Debugging Support

Ability to allow debugging tools to implement generic object browsers by utilizing the standardized information and data model. Example: NRF Bluetooth Smart sniffer from Nordic Semiconductor [nRF-Sniffer].

o Translation

The ability for gateways and other similar devices to dynamically translate (or map) one data model to another one. An example of this idea can be found in [UDI].

o Runtime Discovery

Allow IoT devices to exchange information, potentially along with the data exchange, to discover meta-data about the data and, potentially, even a self-describing interaction model. An example of such an approach has been shown with HATEOAS [HATEOAS].

5. Translation

One of the targets of interoperability is to create translators between the data models. There are analogies with gateways back in 1985 when they were used to translate between network protocols, eventually IP took over providing interoperability, however we lost some of the features provided by those other protocols. The creation of an equivalent "hub/s" that offer translation between the different data models or data semantics seems one of the ways forward. Some lose of expressiveness due to the translation between models seems also unavoidable.

When it comes to translation two different distinctions appear: translating data between data models and translating data models. The first one implies doing the translation at runtime while the second performs one translation between the data models one time, like translating a YANG model to a RAML/JSON one. Indeed, for every IM multiple DMs could be translated.

In a sense these distinctions affect as to when the translation is performed. It can be done at "design time" when the information model is done, it can be done at runtime for a concrete data model and it can be done depending n the actual serialization.

Yet another distinction will appear depending on the requirements from the application protocols, RPC-style ones might require a slightly different DM than REST ones for similar operations, for example SNMP-traps could be similar to CoAP-Observations but not quite the same. It is easier to translate between systems that follow the same architecture/design pattern than across architectures, full translation might not even be possible (e.g. stateless vsstateful systems).

Translation of models script to translate XML to YANG Translation of serialized data, on a hub in order to normalize it to other data model.

6. Dealing with change

A large part of the workshop was dedicated to the evolution of devices and server side applications. Multiple of the participating groups have defined data formats for data representation, however interactions between devices and services and how their relationship evolves over time is more related to the interaction model.

There are various approaches to it. In the most primitive case, a developer will use a description of an API and implement whichever are the protocol steps. In some cases the information model itself can be used to generate some of the code stubs. Changes of an API imply changes on the clients to upgrade to the new version, which requires some development of new code to satisfy the needs of the new API.

New, approaches imply that the whole interaction could be machine-understandable on the first place with changes happening at runtime. In it, a machine client can discover the possible interactions with a service, adapting to changes as they occur without specific code being developed to adapt to them.

The challenge seems to be to define the human-readable parts as machine-readable. Machine-readable require a shared vocabulary to give meaning to the tags.

These type of interactions are based on the The REST architectural style. the principle is that the device or the endpoint, just needs a single entry point, the server would provide descriptions of the API inband by means of web links and forms.

By defining IoT specific relation types, it is possible to drive interactions through links instead of hardcoding URIs into the client, thus making the system flexible enough for later changes. The definition of the basic hypermedia formats for IoT is still work in progress, however some of the existing mechanism can be reused, such as resource discovery, forms or links.

7. Acknowledgements

We would like to thank all paper authors and participants for their contributions. Due to the large number of position paper submissions we were unfortunately unable to invite every author; we would like to apologize to those that could not attend the workshop.

8. Appendix A: Program Committee

This workshop was organized by the following individuals: Jari Arkko, Ralph Droms, Jaime Jimenez, Michael Koster, Dave Thaler, and Hannes Tschofenig.

9. Appendix B: Accepted Position Papers

1. Jari Arkko, "Gadgets and Protocols Come and Go, Data Is Forever"

2. Carsten Bormann, "Noise in specifications hurts"

3. Benoit Claise, "YANG as the Data Modelling Language in the IoT space"

4. Robert Cragie, "The ZigBee Cluster Library over IP"

5. Dee Denteneer, Michael Verschoor, Teresa Zotti, "Fairhair: interoperable IoT services for major Building Automation and Lighting Control ecosystems"

6. Universal Devices, "Object Oriented Approach to IoT Interoperability"

7. Bryant Eastham, "Interoperability and the OpenDOF Project"

8. Stephen Farrell, Alissa Cooper, "It's Often True: Security's Ignored (IOTSI) - and Privacy too"

9. Christian Groves, Lui Yan, ang Weiwei, "Overview of IoT semantics landscape"

10. Ted Hardie, "Loci of Interoperability for the Internet of Things"

11. Russ Housley, "Vehicle-to-Vehicle and Vehicle-to-Infrastructure Communications"

12. Jaime Jimenez, Michael Koster, Hannes Tschofenig, "IPSO Smart Objects"

13. David Jones, IOTDB - "Interoperability Through Semantic Metastandards"

14. Sebastian Kaebisch, Darko Anicic, "Thing Description as Enabler of Semantic Interoperability on the Web of Things"

15. Achilleas Kemos, "Alliance for Internet of Things Innovation Semantic Interoperability Release 2.0, AIOTI WG03 - IoT Standardisation"

16. Ari Keraenen, Cullen Jennings, "SenML: simple building block for IoT semantic interoperability"

17. Dongmyoung Kim, Yunchul Choi, Yonggeun Hong, "Research on Unified Data Model and Framework to Support Interoperability between IoT Applications"

18. Michael Koster, "Model-Based Hypertext Language"

19. Matthias Kovatsch, Yassin N. Hassan, Klaus Hartke, "Semantic Interoperability Requires self describing Interaction Models"

20. Kai Kreuzer, "A Pragmatic Approach to Interoperability in the Internet of Things"

21. Barry Leiba, "Position Paper"

22. Marcello Lioy, "AllJoyn"

23. Kerry Lynn, Laird Dornin, "Modeling RESTful APIs with JSON Hyper-Schema"

24. Erik Nordmark, "Thoughts on IoT Semantic Interoperability: Scope of security issues"

25. Open Geospatial Consortium, "OGC SensorThings API: Communicating "Where" in the Web of Things"

26. Jean Paoli, Taqi Jaffri, "IoT Information Model Interoperability: An Open, Crowd-Sourced Approach in Three Parallel Parti"

27. Joaquin Prado, "OMA Lightweight M2M Resource Model"

28. Dave Raggett, Soumya Kanti Datta, "Input paper for IAB Semantic Interoperability Workshop"

29. Pete Rai, Stephen Tallamy, "Semantic Overlays Over Immutable Data to Facilitate Time and Context Specific Interoperability"

30. Jasper Roes, Laura Daniele, "Towards semantic interoperability in the IoT using the Smart Appliances REFerence ontology (SAREF) and its extensions"

31. Max Senges, "Submission for IAB IoT Sematic Interoperability workshop"

32. Bill Silverajan, Mert Ocak, Jaime Jimenez, "Implementation Experiences of Semantic Interoperability for RESTful Gateway Management"

33. Ned Smith, Jeff Sedayao, Claire Vishik, "Key Semantic Interoperability Gaps in the Internet-of-Things Meta-Models"

34. Robert Sparks and Ben Campbell, "Considerations for certain IoT based services"

35. J. Clarke Stevens, "Open Connectivity Foundation oneIoTa Tool"

36. J. Clarke Stevens, Piper Merriam, "Derived Models for Interoperability Between IoT Ecosystems"

37. Ravi Subramaniam, "Semantic Interoperability in Open Connectivity Foundation (OCF) - formerly Open Interconnect Consortium (OIC)""

38. Andrew Sullivan, "Position paper for IOTSI workshop"

39. Darshak Thakore, "IoT Security in the context of Semantic Interoperability"

40. Dave Thaler, "IoT Bridge Taxonomy"

41. Dave Thaler, S"ummary of AllSeen Alliance Work Relevant to Semantic Interoperability"

42. Mark Underwood, Michael Gruninger, Leo Obrst, Ken Baclawski, Mike Bennett, Gary Berg-Cross, Torsten Hahmann, Ram Sriram, "Internet of Things: Toward Smart Networked Systems and Societies"

43. Peter van der Stok, Andy Bierman, "YANG-Based Constrained Management Interface (CoMI)"

10. Appendix C: List of Participants

- Andy Bierman, YumaWorks
- Carsten Bormann, Uni Bremen/TZI
- Ben Campbell, Oracle

o Benoit Claise, Cisco

o Alissa Cooper, Cisco

o Robert Cragie, ARM Limited

o Laura Daniele, TNO

o Bryant Eastham, OpenDof

o Christian Groves, Huawei

o Ted Hardie, Google

o Yonggeun Hong, ETRI

o Russ Housley, Vigil Security

o David Janes, IOTDB

o Jaime Jimenez, Ericsson

o Shailendra Karody, Catalina Labs

o Ari Keraenen, Ericsson

o Michael Koster, SmartThings

o Matthias Kovatsch, Siemens

o Kai Kreuzer, Deutsche Telekom

o Barry Leiba, Huawei

o Steve Liang, Uni Calgary

o Marcello Lioy, Qualcomm

o Kerry Lynn, Verizon

o Mayan Mathen, Catalina Labs

o Erik Nordmenk, Arista

o Jean Paoli, Microsoft

o Joaquin Prado, OMA

- o Dave Raggett, W3C
- o Max Senges, Google
- o Ned Smith, Intel
- o Robert Sparks, Oracle
- o Ram Sriram, NIST
- o Clarke Stevens
- o Ram Subramanian, Intel
- o Andrew Sullivan, DIN
- o Darshak Thakore, Cablelabs
- o Dave Thaler, Microsoft
- o Hannes Tschofenig, ARM Limited
- o Michael Verschoor, Philips Lightning

11. Informative References

[Allseen-Plugin]
Rockwell, B., "Using the AllJoyn Studio Extension", https://channel9.msdn.com/Blogs/Internet-of-Things-Blog/Using-the-AllJoyn--Studio-Extension , 2016.

[HATEOAS] Kovatsch, M., "Semantic Interoperability Requires Selfdescribing Interaction Models - HATEOAS for the Internet of Things", Proceedings of the IoT Semantic Interoperability Workshop 2016, 2016.

[IOTSIAG] IAB, "IoT Workshop for Semantic Interoperability (IOTSI) - Agenda and Slides", 2016, <https://www.iab.org/activities/workshops/iotsi/agenda/>.

[IOTSIGIT]
IOTSI, "Github Collaborative Repository", 2016, <https://github.com/iotsi>.

[IOTSIWS] IAB, "IoT Workshop for Semantic Interoperability (IOTSI) 2016 - Main Page and Position Papers", 2016, <https://www.iab.org/activities/workshops/iotsi/>.

[LWM2M-Schema]
OMA, "OMA LWM2M XML Schema", http://technical.openmobilealliance.org/tech/profiles/LWM2M.xsd , 2016.

[OMA-Editor]
OMA, "OMA LWM2M Object and Resource Editor", http://dev_devtoolkit.openmobilealliance.org/OEditor/ , 2016.

[OMNA] OMA, "OMNA Lightweight M2M (LWM2M) Object & Resource Registry", http://technical.openmobilealliance.org/Technical/technical-information/omna/lightweight-m2m-lwm2m-object-registry , 2016.

[PYANG] Bjorklund, M., "An extensible YANG validator and converter in python", https://github.com/mbj4668/pyang , 2016.

[RFC3444] Pras, A. and J. Schoenwaelder, "On the Difference between Information Models and Data Models", RFC 3444, DOI 10.17487/RFC3444, January 2003, <http://www.rfc-editor.org/info/rfc3444>.

[UDI] Kohanim, M., "Object Oriented Approach to IoT Interoperability", Proceedings of the IoT Workshop for Semantic Interoperability (IOTSI), 2016.

[nRF-Sniffer]
Nordic Semiconductor, "nRF Sniffer - Smart/Bluetooth low energy packet sniffer", https://www.nordicsemi.com/eng/Products/Bluetooth-Smart-Bluetooth-low-energy/nRF-Sniffer , 2016.

Authors' Addresses

Jaime Jimenez
Ericsson

Email: jaime.jimenez@ericsson.com

Hannes Tschofenig
ARM

Email: hannes.tschofenig@arm.com

Dave Thaler
Microsoft

Email: dthaler@microsoft.com