Survey of IoT Platforms

The Web of Things is intended to be an abstraction layer for the IoT so we need to ensure that it is an effective fit to a broad range of commercial IoT platforms

Dave Raggett <dsr@w3.org>

W3C champion for the Web of Things
Web of Things

• By analogy to the Internet as an abstraction layer that enables services over different networks and technologies
• Web of Things seeks to enable application platforms and open markets of services based upon an abstraction layer for the IoT
• Based upon Linked Data as a lingua franca for data and data models
Linked Data

- A *lingua franca* for data and metadata
  - Basis for relating data and metadata in different formats and data models
- Concepts and relationships are given globally unique identifiers using Web addresses
- These addresses can be used to obtain further information enabling a Web of Linked Data
- W3C has a mature suite of standards
  - e.g. OWL ontology language, SPARQL query language (analogous to SQL)

Linked Data Service

Linked Data makes it easy to combine distributed sources of information

UK Companies House
Linked Data* for Integration

- Expose existing information sources via Linked Data front end that hides the different internal data formats
- Provide Linked Data models for each source
- SPARQL endpoints for remote queries
- RDF-S and OWL for ontologies
- Shape rules for validation and transformation of RDF graphs
- Big data with Linked Data streams
- Combine with access control and terms and conditions of use

* See W3C’s Linked Data page
Survey of IoT Platforms

- Work funded by the European Commission through the Create-IoT Coordination and Support Action
  - Project ID: 732929, part of IoT-02-2016 - IoT Horizontal activities
- Based upon publicly available information for a broad range of IoT platforms
  - Not restricted to IoT sensors and actuators
  - Web of things encompasses abstract entities
    - e.g. Virtual devices, Web services, etc.
- Status – ongoing
- Will be published as a Create-IoT report
Coverage

• So far I have studied
  – OCF OIC 1.1 specifications
  – oneM2M
  – BACnet
  – ECHONET
  – Smart Device Template (HGI)
  – Basic REST over HTTP and CoAP

• Looking for information on other platforms
  – OPC-UA
  – Hypermedia controls
  – Web services

• This survey focuses on the information model

• A further survey is planned for security
Methodology

• Read the IoT platform specifications and look for common patterns
• Examine requirements for the objects exposed to applications
  – Where these objects are decoupled from the underlying IoT platforms, communication patterns, protocols and data formats
• Attempt to reverse engineer Linked Data models for the capabilities defined by each IoT platform
• Express the models in JSON and Linked Data
  – Plain JSON* using a context free grammar
  – Turtle format for Linked Data
• Show case with an HTML based model browser

* Easier to understand than JSON-LD + JSON Schema
OCF

- OIC 1.1 specifications
  - RESTful interface over CoAP and HTTP
  - Specification uses Restful API Markup Language (RAML)
    - Text based, but not markup in the sense of XML
    - Supplemented with tables for property definition and CRUDN behaviour
- OCF define devices in terms of a composition of resources
  - See Smart Home Device specification and Resource Type specification
- Each resource is defined in RAML
  - Identified by a resource type, e.g.
    - oic.r.sensor.acceleration
  - Can import shared definitions for resource types
  - RAML is not used at run-time
  - RAML is verbose with a high degree of predictability
    - I was able to regenerate the RAML from concise thing descriptions using simple heuristics
- OCF specifications available from
  - https://openconnectivity.org/resources/specifications
OCF OIC 1.1 release

• 63 Resources
  – Acceleration sensor
  – Activity Count
  – Altimeter
  – Atmospheric Pressure
  – Air Flow
  – Audio Controls
  – Auto Focus
  – Auto Document Feeder
  – Auto White Balance
  – Battery
  – Binary Switch
  – Brightness
  – ...
  – Water Sensor
  – Weight

• 15 Devices
  – Air Conditioner
  – Air Purifier
  – Window Blind
  – Camera
  – Dishwasher
  – Door
  – Dryer
  – Fan
  – Garage Door
  – Light
  – Oven
  – Printer
  – Multifunction Printer
  – Receiver
  – Refrigerator

See: http://www.w3.org/WoT/demos/td2ttl/oic.html
oneM2M

- RESTful interfaces over CoAP, HTTP and MQTT
- Defines devices in terms of a composition of modules
- Modules are formalized in XML
- Home Appliances and Information Models
  - Defines a large set of devices and modules
  - Broadly speaking similar in coverage to OCF
  - But not compatible ...
  - Guidelines on interworking with LwM2M, AllJoyn and OIC
- See also the base ontology and the SAREF ontology
- oneM2M specifications available from
  - http://www.onem2m.org/technical/published-documents
oneM2M

• 40 Modules
  – Alarm Speaker
  – Audio Video Input
  – Audio Volume
  – Battery
  – Binary Switch
  – Bio Electrical Impedance Analysis
  – Boiler
  – Brightness
  – Clock
  – Colour
  – Colour Saturation
  – ...
  – Wind

• 13 Devices
  – Air Conditioner
  – Clothes Washer
  – Electric Vehicle Charger
  – Light
  – Micro Generation
  – Oven
  – Refrigerator
  – Robot Cleaner
  – Smart Electric Meter
  – Storage Battery
  – Television
  – Thermostat
  – Hot Water

See: [http://www.w3.org/WoT/demos/td2ttl/m2m.html](http://www.w3.org/WoT/demos/td2ttl/m2m.html)
• Building automation
  – Lighting, Heating, ventilation and air conditioning, elevators, emergency lighting & alarms
• Core specs define information model, metadata and protocols
  – Standardised as ISO 164845 and ISO 16484-6
  – BACnet objects are a collection of attributes called “properties”
  – BACnet objects correspond to Web of Things properties
    • 54 object types in BACnet 2012 specification
  – BACnet properties correspond to Web of Things metadata
    • e.g. object names and IDs, high and low limits for values
• Vendors can extend this to suit their devices
  – Need to look at vendor documentation for device definitions
• Essentially properties, actions, events and a lot of metadata
  – Information on whether device is in or out of service, and its reliability
  – 16 level priority mechanism for resolving conflicting property updates
• See Michael Newman’s book
  – “BACnet: the global standard for building automation and control networks”
    • The eBook is much cheaper than the ISO standard!
• Home automation and personal health
  – Devices with properties, events and metadata
  – No actions that I’ve found so far, but …
  – Includes a vocabulary for describing homes
    • Living room, dining room, kitchen, bathroom, lavatory, wash room, passageway, room, stairway, front door, store room, garden, veranda/balcony

• ECHONET specifications available from
  – https://echonet.jp/spec_en/
ECHONET Shared Properties (22)

- Operation Status
- Installation Location
- Standard Version
- Fault Status
- Fault Description
- Manufacturer Code
- Business Facility
- Product Code
- Production Number
- Production Date
- Property Map
- Identification Number
- Manufacturer’s Fault Code
- Current Limit Setting
- Power-Saving Operation Setting
- Remote Control Setting
- Cumulative Operating Time
- Current Time
- Current Date
- Measured Instantaneous Power
- Measured Cumulative Power
- Power Limit Setting
ECHONET Sensors (44)

- Gas Leak
- Crime Prevention
- Emergency Button
- First-Aid
- Earthquake
- Electric Leak
- Human Detection
- Visitor
- Call
- Condensation
- Air Pollution
- Oxygen
- Illuminance
- Sound
- Mailing
- Weight
- Temperature
- Humidity
- Rain
- Water Level
- Bath Water Level
- Bath Heating Status
- Water Leak
- Water Overflow
- Fire
- Cigarette Smoke
- Carbon Dioxide
- Gas
- VOC
- Differential Pressure Loss
- Air Speed
- Odor
- Flame
- Electric Energy
- Current Value
- Water Flow
- Micro-Motion
- Passage
- Bed Presence
- Open/Dose
- Activity Amount
- Human Body Location
- Snow
- Air Pressure
ECHONET Device Groups

- Air Conditioners (10)
  - Home/commercial, ventilation fan, air conditioner fan, air cleaner, humidifier, electric heater, fan, indoor/outdoor, electric storage heater

- Housing/Facilities related (31)
  - Blind/shade, shutter, rain sliding door, ..., buzzer, electric vehicle charger

- Cooking/Household related (10)
  - Electric thermos, refrigerator, microwave, cooker, rice cooker, ..., washer/dryer, outdoor unit

- Health related (5)
  - Weighing machine, thermometer, blood pressure, blood sugar, body fat

- Management Operation related (3)
  - Switch, JEM-A/HA controller, DR event controller

- Audio Visual related (4)
  - Display, television, audio, network camera

Frequent use of enumerations with binary codes, e.g. clean rinsing = 0x29
• Defined by the Home Gateway Initiative
  – XML Schema for smart home devices, see
    • https://github.com/Homegatewayway/RWD050-public
• Devices defined as a set of modules
• Modules define data, actions and events
• Data is HGI’s name for property
• Device info metadata
REST over CoAP and HTTP

- Roy Fielding’s representational state transfer design pattern for stateless protocols
- Resources identified by URLs
- Core methods for operations
  - Create, Read, Update, Delete and Notify
  - Internet media types for content
- Plus means to embed metadata in protocol headers
  - Link relations as basis for discovery
- Used as building blocks for IoT platforms like OCF
- Related to formats for describing RESTful APIs, e.g.
  - RAML
  - Open API Specification (previously Swagger)
  - Hydra: Hypermedia-driven Web APIs
  - WADL
  - Slate
Results
Emergent Requirements

• Strong support for objects with properties, actions and events

• Object model for things
  – Common data types
    • Boolean, string, integer, number
  – Less common data types
    • Enumerations, unions, vectors, ordered/unordered collections
    • Application types in terms of predefined types
    • Things as first class types
    • Compound (nested) properties
    • Compound arguments for actions and events

• Metadata, e.g.
  • Min, max for numbers and integers
  • Read-only or read-write
  • Required and optional
    – Properties and action arguments
  • Units of measure including scale factors
  • Whether static or changeable
  • Cardinality for collections

• Metadata for given IoT platforms

• Mapping of enumerated values, property names etc. to IDs
  • Resilience to changes in models
    – Google protocol buffers

Web of Things should avoid protocol specific data representations, e.g. bit fields
Syntactic Modularity

• Specifications like to define things in a modular way
  – Reusable definitions, e.g. enumerations
  – A set of interfaces (e.g. oneM2M’s modules)
    • Some set of properties, actions, events, metadata and named type definitions
  – A means to combine this when defining devices
    • e.g. refrigerator with freezer and cooler compartments using common interface models for sensors and controls
  – Syntactic modularity analogous to C++ #include
    • At the level of a thing or a thing’s properties
Semantic Modularity

• Not used by any of the IoT platforms covered so far in the survey
  – Useful for semantic search and semantic interoperability
• For each device state what semantic classes it conforms to
• Each semantic class defines requirements in terms of
  – other semantic classes
  – properties, actions, events, metadata
• Can be validated to check for semantic interoperability
  – Ensuring that services share the same meaning, interfaces and data types
• Can also be used abductively to generate a thing description from a set of semantic classes and additional constraints
  – Involves making guesses based upon past experience
Open Questions

• Is it worth extending the survey to cover the communication patterns supported by different IoT platforms and the use cases that motivate these patterns?
  – This would help with respect to guidance on how to bind the Web of thing to particular IoT platforms

• The need to examine what kinds of IoT platform metadata is needed as part of the application contract?
  – This should be sufficient to meet the means for the application platform to make the connection
  – It can also provide a basis for negotiating quality of service
  – But risk of tightly coupling apps to particular IoT platforms
Next Steps

• I plan to flesh out this survey as a CreateIoT report whilst also extending it to a broader range of IoT platforms
  – My highest priority is OPC-UA
  – Other targets include Apple HomeKit, Android Things, OMA’s LwM2M, OMG’s DDS, Hypermedia Controls and a wide range of Web services
  – Any suggestions for platforms to look at?
• To be complemented with a companion survey on end to end security across different platforms and through the lifecycle
Thank you

Questions?