

IoT Extensions for schema.org

Community Teleconference

January 20, 2020

Agenda

- Agenda review
- Announcements – T2TRG Workshop on Data Models and One Data Model + WISHI Hackathon
- Recent developments – Project CHIP
- Proposal for adding a thing class
- Availability of iotschema4Node-RED over npm
- Hosting and schema.org extension
- AOB

Announcements

- W3C WoT Online Virtual F2F March 16-18
- T2TRG Workshop March 20 at IETF 107 Vancouver
 - Technical review of SDF language (OneDM)
- WISHI Hackathon March 21 and 22 at IETF 107
 - Semantic Proxy using W3C WoT and iotschema annotation from OneDM definitions
- OneDM F2F late April, Qualcomm in San Diego
- W3C WoT Helsinki June 6-11 with T2TRG

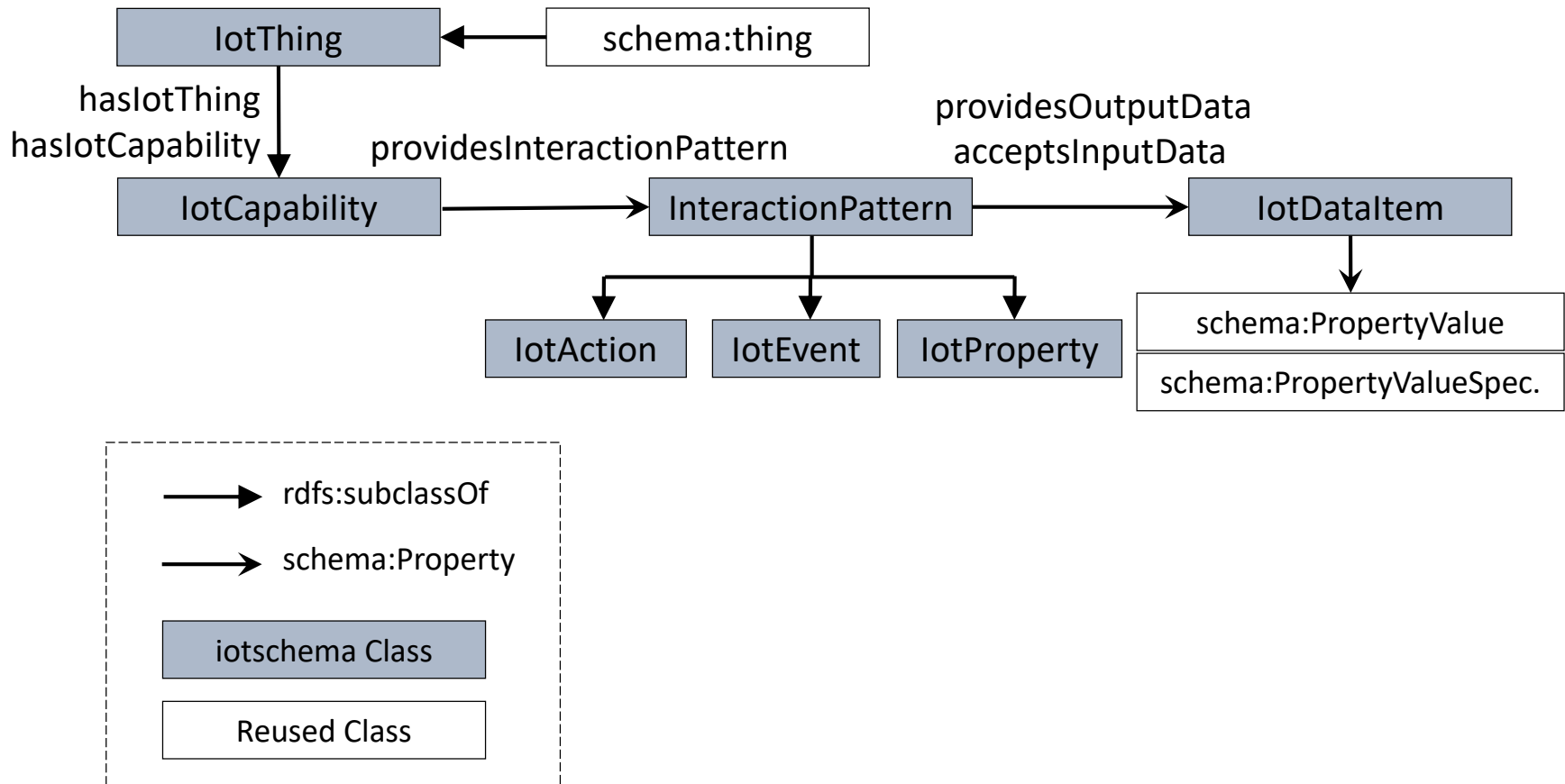
Project CHIP

- Google and Apple joined Zigbee Alliance to create a new interoperable network standard for connected homes – Project Connected Home over IP
- Deliver a standard, open source reference stack, and certification program for interoperable devices
- What it means to iotschema - standardized data models for connected home devices + Event, Action, Property model + open source license
- Still need to address system level modeling with location, context , behavior + application domains

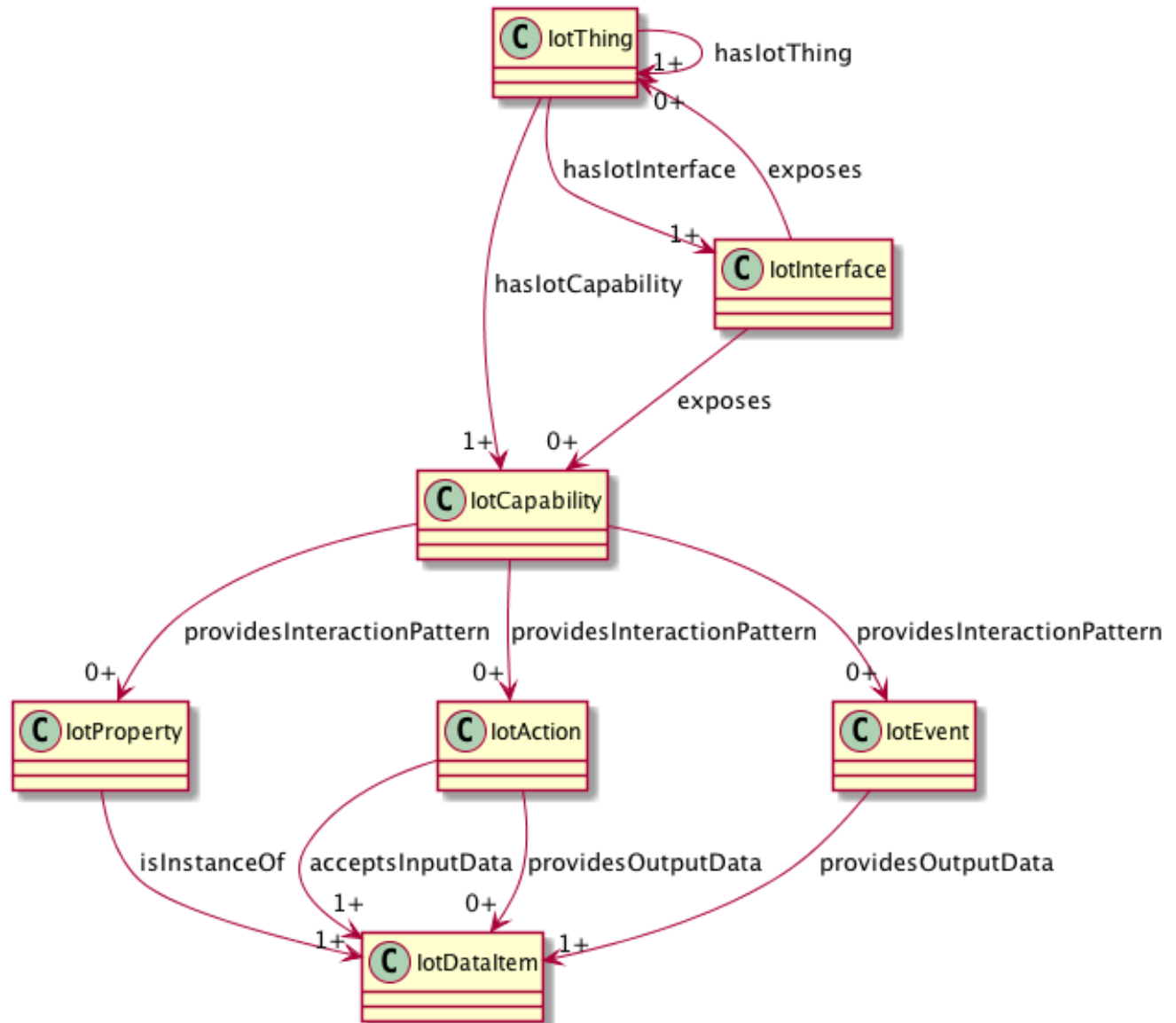
Thing Class

- Encapsulate reusable Capabilities
 - On/Off with state Property, Commands, and Events
 - Compose Air Conditioner Thing from OnOff, Mode, Speed, etc. as reusable Capabilities
- Reusable compositions of Capabilities
 - An Outlet unit for a multi-outlet strip
 - Each Outlet has OnOff, Energy Monitor, Overcurrent and Overtemperature protection Capabilities
 - Multiple Outlets are composed into an outlet strip
 - Outlet unit can be a Thing
 - Outlet Strip can also be a Thing

schema.org IoT Extension Meta Model with Thing Class



UML



iotschema4Node-RED

iot.schema.org for Node-RED

Semantic Interoperability

- iot.schema.org embedded in Node-RED tool
- Easies use of semantics for WoT developers
- Simplify creation of applications with W3C WoT
- Demonstrates semantic discovery and processing
- Handsome tool for WoT PlugFests

Publishing iotschema Nodes to npm

- iotschema nodes are currently being adapted for publication on the npm repository
- This will allow iotschema Node-RED developers to easily find and access nodes without having to go via the GitHub route.
- Currently available:
 - 17 packages
 - Migration for packages A-C completed
 - D-W in progress

How to use npm with iotschema nodes

- Users can simply issue the ``npm install`` command from their Node-RED environment
 - Example: `npm install @iotschema/binaryswitch`
- Nodes are hosted on the npm repository and are browsable on npmjs.com

@iotschema/activeenergyexported

The electric energy derived from active power over a period of time, and exported (delivered).

iotschema node node-red activeenergyexported



iotschema published 1.0.0-beta • 3 days ago

@iotschema/activeenergyexported

1.0.0-beta • Public • Published 3 days ago



Readme



Explore BETA



0 Dependencies



0 Dependents



1 Versions

"Unable to find a readme for @iotschema/activeenergyexported@1.0.0-beta"

Keywords

iotschema node node-red activeenergyexported

Install

```
> npm i @iotschema/activeenergyexported
```

± Weekly Downloads

21

Version

1.0.0-beta

License

none

p
q
m

Publishing iotschema Nodes to npm – Grouping

- One umbrella module (iotschema-capability) to install all nodes
- More granular groups to install only specific types
 - Example:

@iotschema/ambientair-capability

|

- └ @iotschema/nitrogenconcentration
- └ @iotschema/oxygenconcentration
- └ @iotschema/carbondioxideconcentration
- └ @iotschema/argonconcentration

Schema.org Integration

- Class names Event, Action, Property conflict
- iotschema has diverse semantic types for objects, schema.org has diverse property types
- Property types could be synthesized from objects but...
- iotschema will potentially define hundreds of types for physical quantities (temperature, humidity, voltage, acceleration...), control affordances (open/close, brightness, color control, camera controls, operating modes...), and features of interest (rooms, machines...)

Schema.org Integration

- The WoT use case is based on annotation consisting of RDF @type statements that point to URIs of defined terms for specialized types that conform to the classes in the meta-model
- These meta-model classes would only add about 6 new property types to schema.org
 - iotCapability, iotEvent, iotAction, iotProperty, iotData, iotFeatureofInterest
 - new types like iotInterface, iotThing, etc. as needed

Schema.org Integration

- There is a potential example pattern in schema.org
 - MedicalEntity, with about 7 property types
- Likewise, an instance of IoT Schema would contain some set of iotCapability, iotAction, iotProperty, iotEvent, iotFeatureOfInterest properties
- Specialization of iot types would happen at the next level in the graph – hosted in a separate namespace
 - URIs that point to accepted specialized definitions in one or more specialized namespaces
 - lighting controls, thermostats, etc. that conform to the base types but have their own properties

Conclusion

- Summary
- AOB