WISHI Interop Planning and Implementation Guidance

Michael J Koster

February 19, 2018

Contents

- Extrinsic Semantics => Feature of Interest
- Interop Planning
- Directory and Repository requirements
- Review of Wiki Materials

Semantics to describe the context of a connected thing

- **Feature of Interest** is a general concept in the O&M ontology used to denote some physical feature from which an observation was obtained (in SSN, a sensor observes a property of a FoI)
- We can generalize this concept to enable fine-grained definitions that can apply to FoI instances, e.g. "*kitchen + illumination*", "*outboard + spoiler*"
- We can then bind instances of Capabilities and Interactions of connected things to the Features of Interest they sense (or actuate): "*kitchen illumination + color temperature* ", "*outboard spoiler + position*"
- The rudimentary pattern is already in the iotschema core definition: *hasFeatureofInterest, actsOn, isActedOnBy*

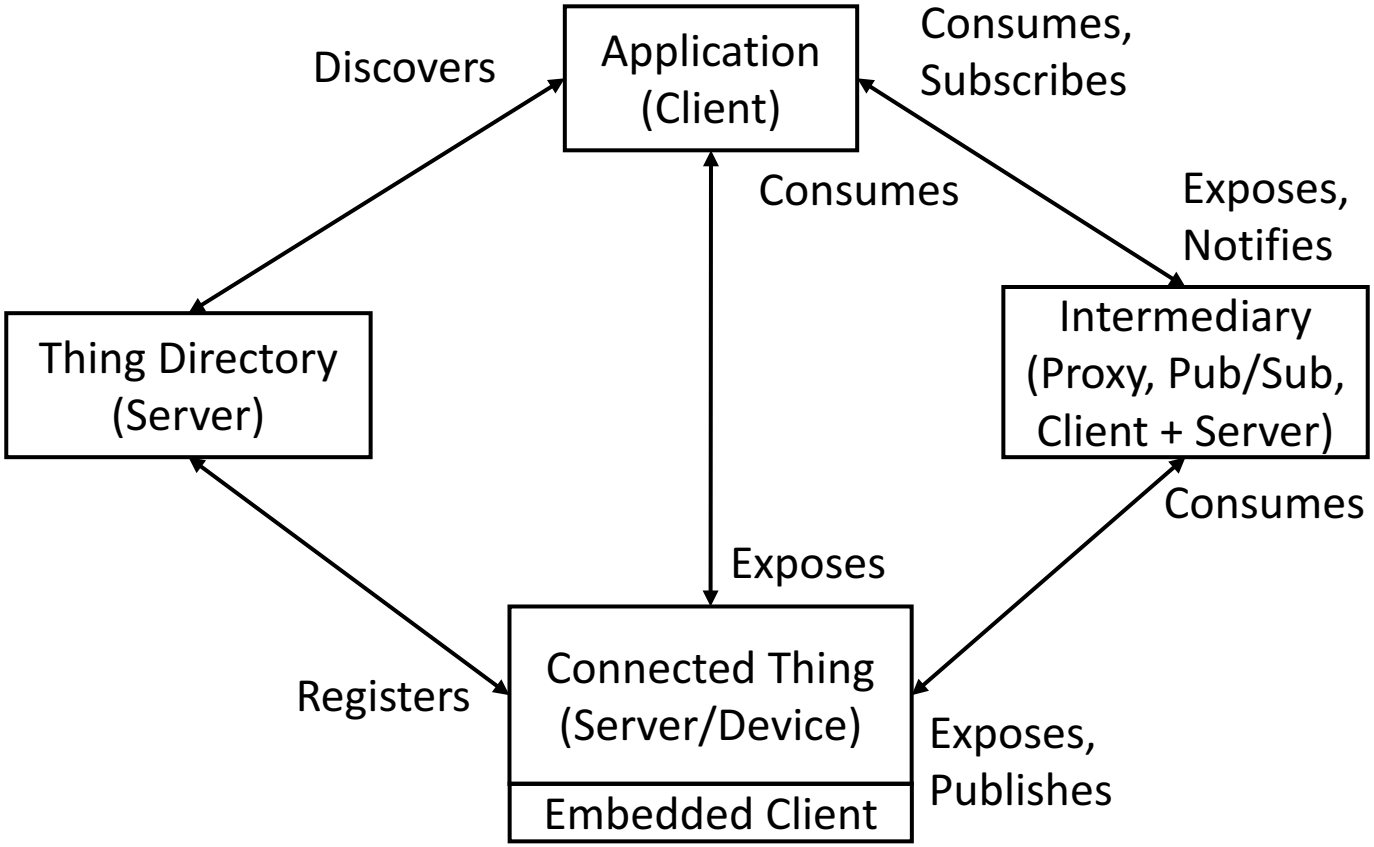
Interop Planning

- Goals and focus
- Schematic and plan
- Preparation Documents
  - Scenarios => Test Cases
  - Participant Questionnaire
  - Interop Online

Goals and Focus

- Explore and test design patterns for Semantic Interoperability and Hypermedia
- Registration and Discovery – What to do
  - Thing Directory/Resource Directory
  - Hyperlinks with Semantic Annotation
- Client adaptation to servers – How to do it
  - Hypermedia controls
  - Gateways and translations

Interop Schematic Diagram – Roles and Interactions



Scenarios and Test Cases

- Scenarios to enable interoperability testing
  - Register things in a directory using a common format
  - Applications discover things that others have registered
  - Application can use information in discovery to learn how to adapt to thing protocols or find a translator
  - Application can use connected things that others have brought, e.g. a motion sensor controlling a light
  - Replace components of an application, e.g. a different Motion Sensor
  - Develop client adaptations to diverse thing protocols
  - Develop gateway translations between diverse protocols

Participant Questionnaire

- For anyone participating whether hands-on or not
- Describe functions and protocols involved in the various roles:
  - Connected Thing (Embedded Client)
  - Application Client
  - Proxy or other Intermediary
  - Directory
- Facilitate discussion and coordination
- Any particular role is optional

Questionnaire - Functionality

- What functionality for the various system roles are you planning to bring?
  - Scenarios – What are the application scenarios being modeled
  - Application and Client
  - Connected Thing and Server
  - Discovery and Directory
  - Intermediary – Proxy or Pub/Sub

Questionnaire - Protocols

- Which Target Protocols do your servers expose?
- How are dynamic notifications on observable resources handled by your servers?
- Which target protocols does your client support?

Questionnaire - Semantic Annotation

- What thing types and Capabilities will be exposed by your servers?
- What interactions and data types will be exposed for each capability?
- How does the client know what things, capabilities, and interactions to look for?
- What application does your client host?

Questionnaire - Directory

- How will you do registration life cycle management?
- What resources and services will you register?
- What client discovery methods will you use?
  - Semantic queries, formats
- Additional semantic annotation about context
  - How will you augment the registrations with additional context metadata

Questionnaire - Intermediary

- What protocols does your proxy consume?
- What protocols does your proxy expose?
- What other protocols are needed to interact with your proxy?
  - To expose things through the proxy?
  - To consume things through the proxy?
- How does your proxy interact with Directories for discovery of consumed things and registration of exposed things?
- How does your proxy remap URIs?

Questionnaire - Security

- What transport security methods are supported, e.g. DTLS?
- What access control features are implemented, e.g. HTTP basic auth, kerberos tokens, ACLs ?
- How is security material obtained and configured?
- What additional services are needed/provided such as Auth Server (AS)?

Questionnaire - Accessibility

- What Accessibility scenario is being modeled?
- What alternate modes of interaction are provided by the client and application?
- What are additional requirements on the Thing Directory, Server and Proxy to support your Accessibility Scenario?

Interop Online

- Provide an online "network" to test interoperability outside of face to face Interop events
- Preview of components that are known in advance, e.g. a Directory implementation
- Most IP protocols will work over a WAN but need some special considerations
- Test LAN-WAN connectivity in Interop events

Directory and Repository

- Directory – Example: WoT Thing Directory
  - Used to register entry points to instances and affordances of connected things
  - Used in Discovery to locate entry points of interesting connected things and services
  - Enables both intrinsic and late binding (extrinsic) annotation to be used for selective discovery
- Repository – Example: iotschema
  - Used to obtain common terminology to annotate instances and affordances of connected things
  - Contains vocabularies and ontological graphs that provide meaningful information about the affordances that use the annotations

Directory Requirements

- Well-known protocols
  - CoAP, HTTP transfer protocols
  - CoRE RD registration and discovery
  - LAN and WAN compatibility (URI resolution, etc.)
- Standard representation formats
  - CoRE Link-Format
  - W3C Thing Description
- Common Vocabulary to describe connected things and their context (External)
- Well-known query mechanisms
  - CoRE RD
  - SPARQL

Repository Requirements

- Well-known place to obtain definitions for terms
- Machine- and Human-readable formats
- Provide meaningful information about terms
- Common Core vocabulary for base interoperability
- Domain-agnostic vocabulary support
- Protocol-agnostic formats
- Permissive licensing for contributions and use of content
- Easy to contribute to and maintain from diverse application domains

Review of Wiki Materials

- General Resources
  - Directory and Repository Requirements
  - Tools and Open Source Projects
  - Standards and Formats
  - Implementation Guidance and Examples
- Interop Planning Documents
  - Overview and Goals
  - Scenarios and Test Cases
  - Participant Questionnaire
  - Interop Online