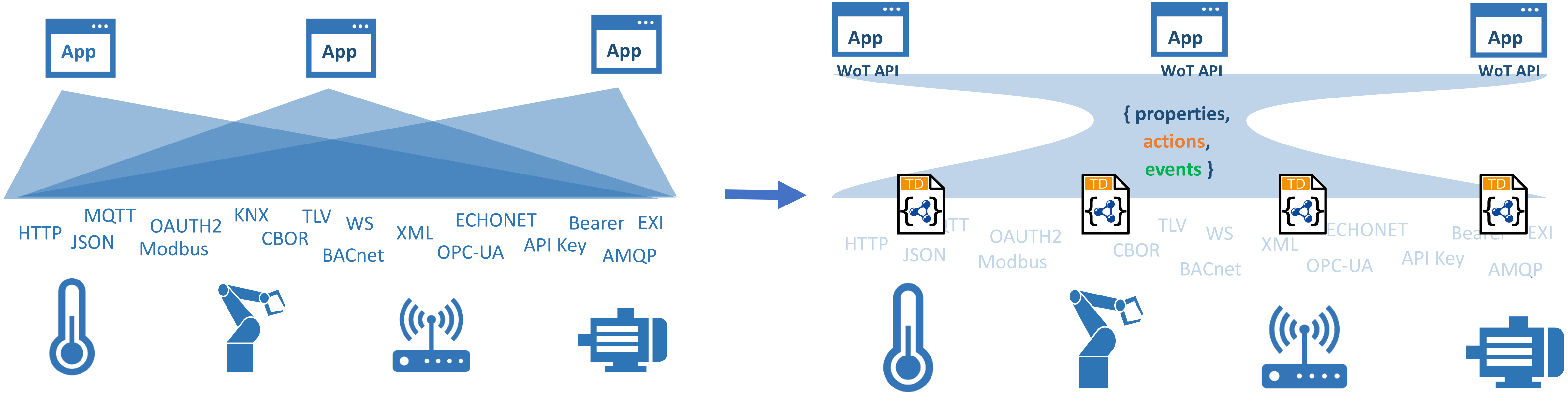
Web of Things Update

Michael McCool

June 2021

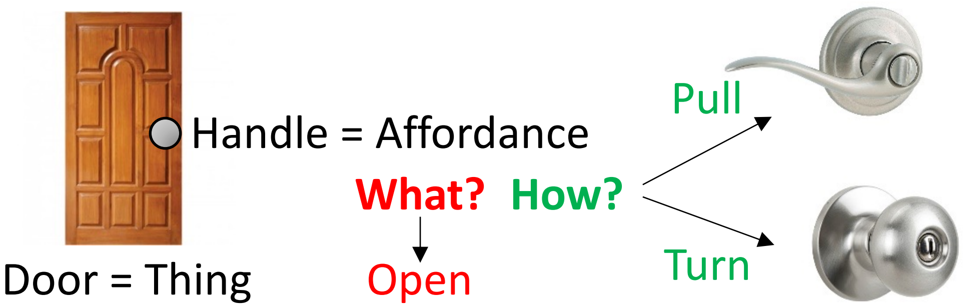
W3C Web of Things (WoT)

- **W3C WoT Working Group goal:** Adapting web technologies to IoT
- **Published:** Thing Description (TD) metadata format
  - TD describes the available interactions (network API) of a Thing
- **In Progress:** TD 1.1 Update, Thing Models, Discovery, Profiles
  - How to obtain TDs?   How to ensure interoperability?



WoT Thing Descriptions

- WHAT the possible choices are
  - Properties
  - Events
  - Actions
- HOW to interact with the Thing
  - Protocol operations and options
  - Data schemas and content types
  - Security requirements



Handle = Affordance

What? How?

Pull

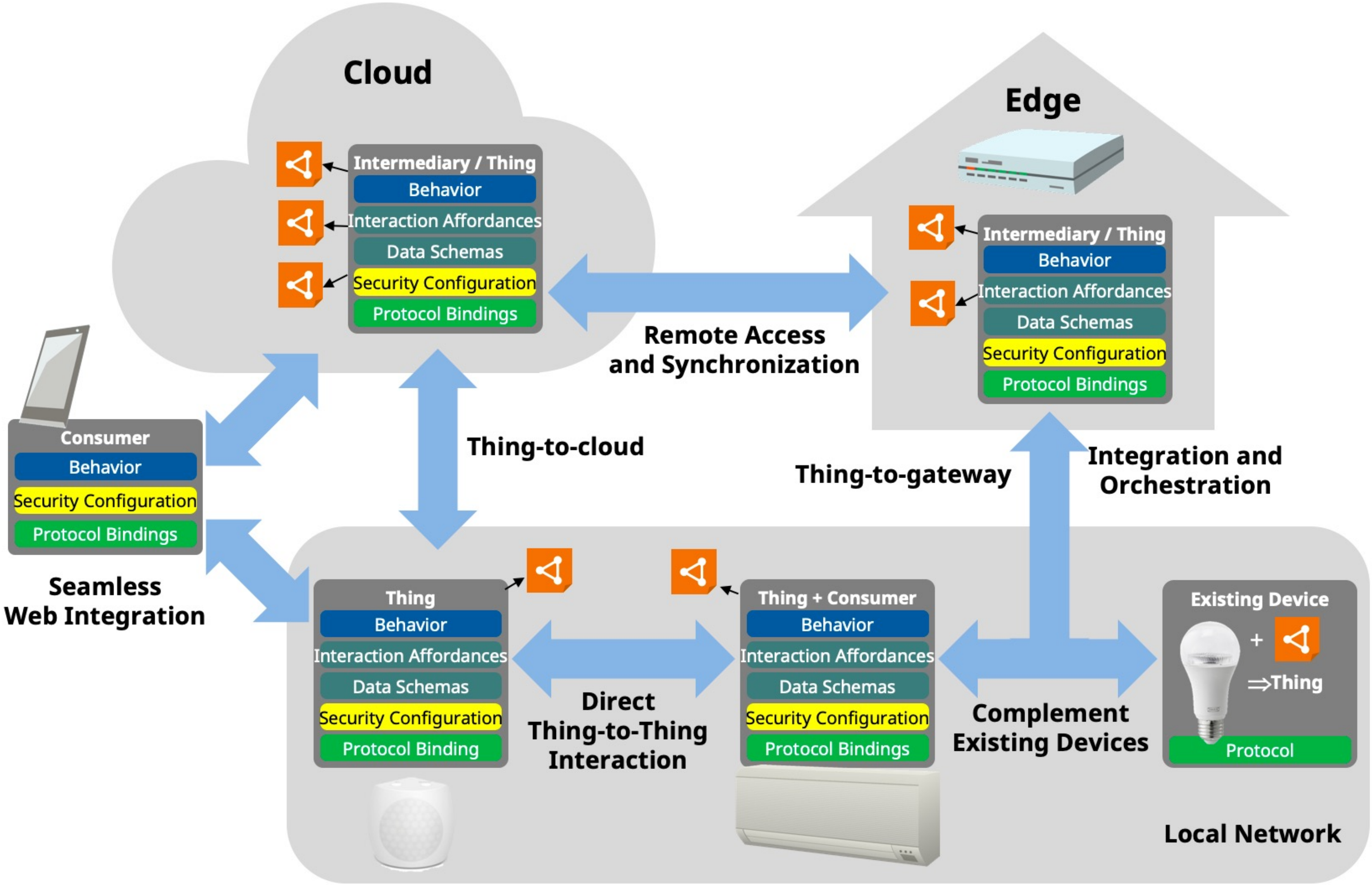
Turn

Open

Door = Thing

```
{
  "@context": [
    "https://www.w3.org/ns/td",
    { "iot": "http://iotschema.org/" }
  ],
  "id": "urn:dev:ops:32473-WoTLamp-1234",
  "title": "MyLEDThing",
  "description": "RGB LED torchiere",
  "@type": ["Thing", "iot:Light"],
  "securityDefinitions": ["default": {
    "scheme": "bearer"
  }],
  "security": ["default"],
  "properties": {
    "brightness": {
      "@type": ["iot:Brightness"],
      "type": "integer",
      "minimum": 0,
      "maximum": 100,
      "forms": [ ... ]
    }
  },
  "actions": {
    "fadeIn": {
      ...
```

Usage Patterns



Current WoT WG Charter Work Items

Architectural Requirements, Use Cases, and Vocabulary

- Understand and state requirements for new use cases, architectural patterns, and concepts.

Link Relation Types:

- Definition of specific link relation types for specific relationships.

Observe Defaults:

- For protocols such as HTTP where multiple ways to implement "observe" is possible, define a default.

Implementation View Spec:

- More fully define details of implementations.

Interoperability Profiles:

- Support plug-and-play interoperabilty via a profile mechanism
- Define profiles that allow for finite implementability

Thing Models:

- Define how Thing Descriptions can defined in a modular way.

Complex Interactions:

- Document how complex interactions can be supported via hypermedia controls.

Discovery:

- Define how Things are discovered in both local and global contexts and Thing Descriptions are distributed.

Identifier Management:

- Mitigate privacy risks by defining how identifiers are managed and updated.

Security Schemes:

- Vocabulary for new security schemes supporting targeted protocols and use cases.

Thing Description Vocabulary:

- Extensions to Thing Description vocabulary definitions.

Protocol Vocabulary and Bindings:

- Extensions to protocol vocabulary definitions and protocol bindings.

New Deliverables

- Thing Description 1.1
  - Canonicalization (and WIP, Signing)
  - Validation levels
  - Thing Model
  - various other extensions, e.g. to security, data schemas, etc.
- Discovery
  - Introductions: DNS-SD, DID, CoRE RD
  - Directory Service: HTTP API for searchable database of TDs
  - Self-Description: .well-known, fetching of TD directly from Things
- Profiles
  - Emphasis on "hub" use-case, http/json
- Use Cases and Requirements (informative document)

Thing Description 1.1: Updates

- Canonicalization and Signing
  - WIP, but proposal is based on JOSE/JWS/JWA (incl. RFC 8037)
  - Can extract parts of a TD to sign using JSONPointer/JSONPath/XPath queries
- Security Scheme Improvements
  - URI Templates
  - Security information in body
  - OAuth "device" flow
- Thing Model
  - TD describes instance, TM describes class
  - Provides templating/parameterization mechanism
  - TD can reference *one* TM using a link
  - TMs can reference or extend other TMs (and *parts* of other TMs)

Discovery: Goals

Capabilities

- Support both local and global/remote discovery (unconstrained by network domain)
- Support "localizable" discovery (constrainable by location)
- Support both "syntactic query" (keywords) and "semantic query" (linked data)
- Support a directory service for searching large repositories of Things
- Support peer-to-peer (self-identifying "smart object") discovery

Privacy-Preserving Architecture

- Respect device and information Lifecycle
- Distribute TDs only to authenticated and authorized users
- Don’t leak private data to unauthorized users
- Don’t leak data that can be used to INFER private information to unauthorized users

Alignment with Existing and Evolving Standards

- IETF CoRE Resource Directories, CoRE Link Format, DID, OGC, WGS84, XPath, …
- Compatible with WoT Scripting API

Discovery: Two-Phase Architecture

Phase 1: Introduction

- “First Contact” Protocol
  - Answers the question: How to initiate discovery from zero knowledge?
- Open
  - Can be accessed with no or limited access controls
  - Based on existing standards, and can be extended to new standards
- Lightweight
  - Does not use significant resources on responder
  - Resistant to Denial of Service attacks
- Provides intentionally limited information
  - Avoid leaking any metadata that can be used to infer private data
  - This includes types of devices, device ids, owners, timestamps, etc.

Phase 2: Exploration

- Authentication and authorization required
- Supports more complex query and filtering capabilities (JSON Path, XPath, SPARQL)
- Provides access to rich metadata (TDs)
- Access controls can limit data returned

Discovery: Status

Introductions

- DNS-SD (including mDNS) – new service names
- CoRE RD – resource types
- DID – endpoint types
- Well-known URLs: to "guess" URL from an IP
- Direct: anything else that returns a URL
- **Note:** link types distinguishing a Directory and a Thing are useful but not mandatory

Exploration

- "Smart Objects": Retrieve TD directly from Thing
- Directory service API: described using a TD
- Provides multiple query types:
  - JSONPath – mandatory
  - XPath – optional
  - SPARQL – optional
- Pagination, etc.

What is a WoT Profile?

- A **WoT Profile** is a normative subset of a *WoT Thing Description* with a normative binding to a selected protocol.
- Profiles guarantee **interoperability** between compliant implementations, multiple profiles are possible.
- The **WoT Profile Specification** defines a **normative** set of *constraints and rules* on the **data model**, **representation format** and **protocol binding**.
- These constraints and rules provide clarifications and make decisions that reduce the complexity for implementers of the WoT standard.
- The rules are prescriptive, to ensure that compliant implementations satisfy the semantic guarantees implied by them.

Profiles: Constraints

Constraints on	*Rationale*	*Example*
vocabulary of Thing Description classes	guaranteed set of metadata fields	Make specific vocabulary terms mandatory, remove others
class relationships	unambiguous structure	limited cardinality, e.g. only one form per operation per interaction affordance.
values of vocabulary terms	simplified processing	Limit the length of characters per string. Always use arrays, where the spec permits a string or an array of strings.
data schemas	simplified processing	Limits on nesting
security	reduced implementation effort	Only a restricted set of security mechanisms
protocol binding	guaranteed protocol semantics	limited protocol(s) and protocol features, Example: predefined mapping of http verbs (GET/PUT) to operation verbs, similar constraints for other protocols.

Profiles: Current Work

- Defining a core/baseline profile with a HTTP binding.
- Identifying constraints and rules on the data model.
- Unambiguous interaction semantics for properties, actions and events.
- Constraints on payload formats.
- Protocol binding semantics, e.g. headers, response codes.
- Security constraints.
- Representation format constraints.

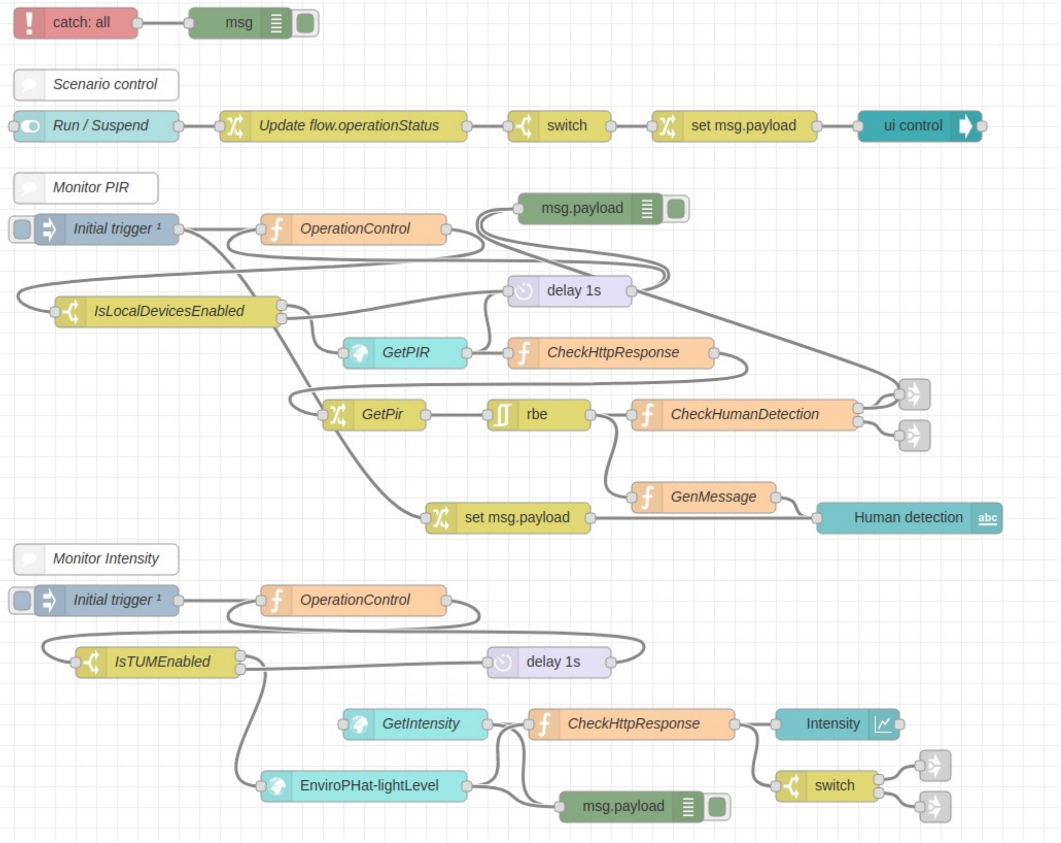
Use Cases – W3C Smart City Workshop

Others

- Law Enforcement
- Parking
- Accessibility
- Traffic and Logistics
- Public Transportation
- Air Quality and Weather
- Cultural Space Mgmt
- Construction Services
- Land Management
- Emergency Services
- Water Management
- Hybrid Ruralization

WoT Orchestration

Node-RED/node-gen



node-wot/Scripting API

```
WoTHelpers.fetch( "coap://localhost:5683/counter" ).then( async (td) => {
  // using await for serial execution (note 'async' in then() of fetch())
  try {
    let thing = await WoT.consume(td);
    console.info( "=== TD ===" );
    console.info(td);
    console.info( "==========" );

    // read property #1
    let read1 = await thing.readProperty( "count" );
    console.info( "count value is" , read1);

    // increment property #1 (without step)
    await thing.invokeAction( "increment" );
    let inc1 = await thing.readProperty( "count" );
    console.info( "count value after increment #1 is" , inc1);

    // increment property #2 (with step)
    await thing.invokeAction( "increment" , {'step' : 3});
    let inc2 = await thing.readProperty( "count" );
    console.info( "count value after increment #2 (with step 3) is" , inc2);

    // decrement property
    await thing.invokeAction( "decrement" );
    let dec1 = await thing.readProperty( "count" );
    console.info( "count value after decrement is" , dec1);

  } catch(err) {
    console.error( "Script error:" , err);
  }

}).catch( (err) => { console.error( "Fetch error:" , err); });
```

Documents and Resources

New/Updated Normative Documents in Draft Status:

- Architecture 1.1: https://github.com/w3c/wot-architecture
- Thing Description 1.1: https://github.com/w3c/wot-thing-description
- Discovery: https://github.com/w3c/wot-discovery
- Profiles: https://github.com/w3c/wot-profile

New/Updated Informative Documents in Draft Status:

- Binding Templates: https://github.com/w3c/wot-binding-templates
- Scripting API: https://github.com/w3c/wot-scripting-api
- Use Cases and Requirements: https://github.com/w3c/wot-usecases

Other Resources:

- Web Site: https://www.w3.org/WoT/

Contacts

https://www.w3.org/WoT

Dr. Michael McCool

Principal Engineer

Intel

Technology Pathfinding

michael.mccool@intel.com

Dr. Sebastian Kaebisch

Senior Key Expert

Siemens

Technology

sebastian.kaebisch@siemens.com

Backup

Image Credits

- Solar Installation Vietnam: By Intel Free Press - https://www.flickr.com/photos/intelfreepress/7169063498/sizes/o/in/photostream/, CC BY 2.0, https://commons.wikimedia.org/w/index.php?curid=28011974
- Telemedicine Consult: By Intel Free Press - https://www.flickr.com/photos/intelfreepress/6948764580/sizes/o/in/photostream/, CC BY 2.0, https://commons.wikimedia.org/wiki/File:Telemedicine_Consult.jpg