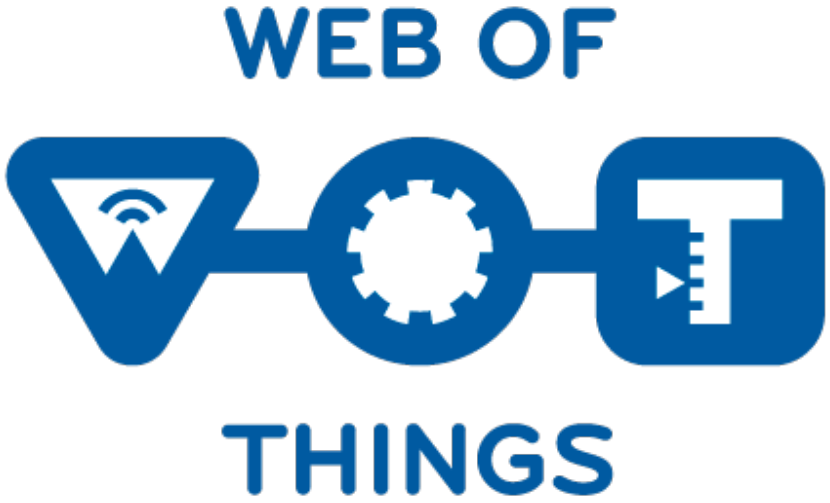


W3C Web of Things Summary and Status Update

Michael McCool

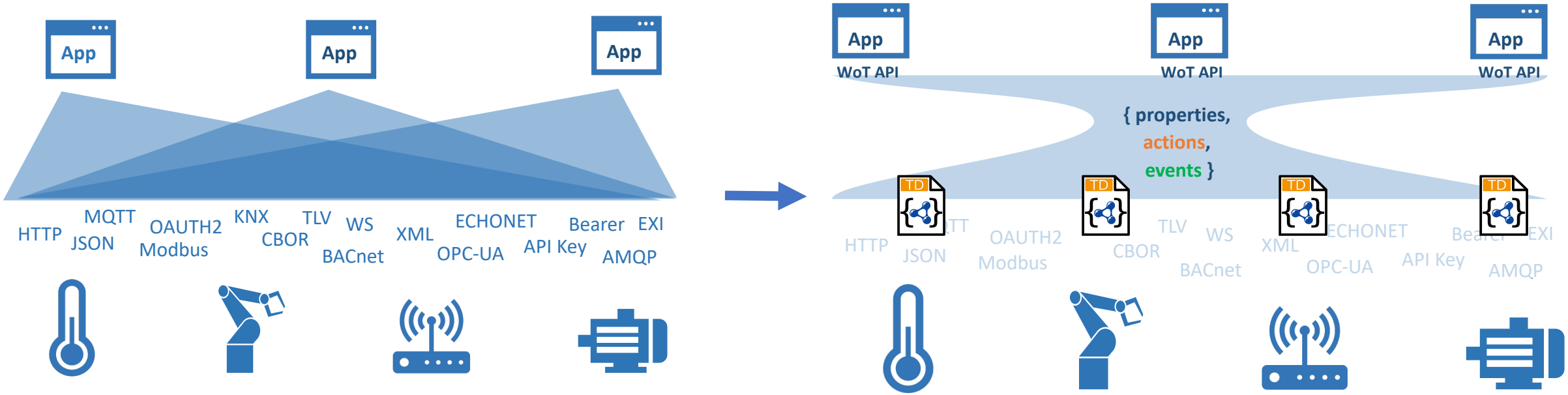
March 2022

Outline

- What is WoT?
  - Applying and extending web standards for IoT
  - Descriptive interoperability: Thing Descriptions
  - Finding Thing Descriptions: Discovery
  - Use Cases and Requirements
- Recent Activity
  - Testfest/Plugfest
  - New Commercial Applications
- Discussion
  - Gaps and Future Work

W3C Web of Things (WoT)

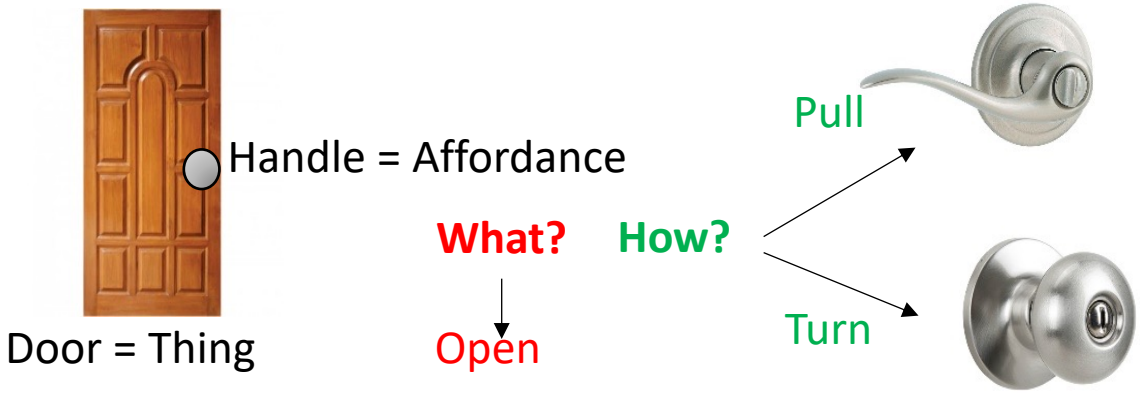
- W3C Working Group goal: Adapting web technologies to IoT
- Already published: Thing Description (TD) metadata format
  - TD describes the available interactions (network API) of a Thing
- New deliverables in progress, including Discovery
  - How does a potential user obtain the TD for a Thing?



Descriptive Interoperability: TDs

WoT Architecture

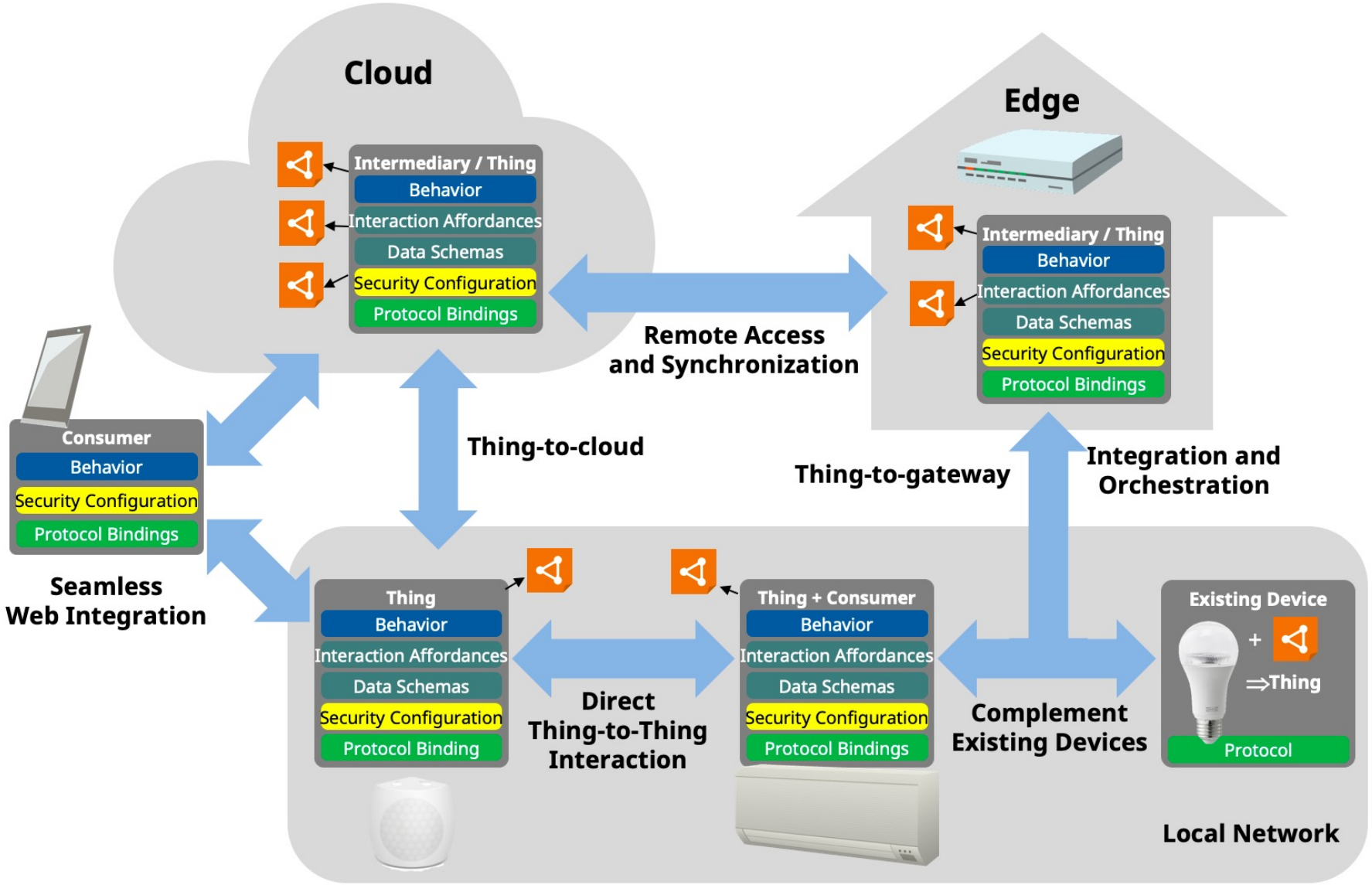
- Constraints
  - "Things" must have a TD
  - Must use URIs, IANA media types, etc.
- Thing Description Affordances
  - Describes WHAT the possible choices are
  - Describes HOW to interact with the Thing



WoT Thing Description (TD)

```
{
  "@context": [
    "https://www.w3.org/2019/wot/td/v1",
    { "iot": "http://iotschema.org/" }
  ],
  "id": "urn:dev:org:32473:1234567890",
  "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 Overview



Discovery

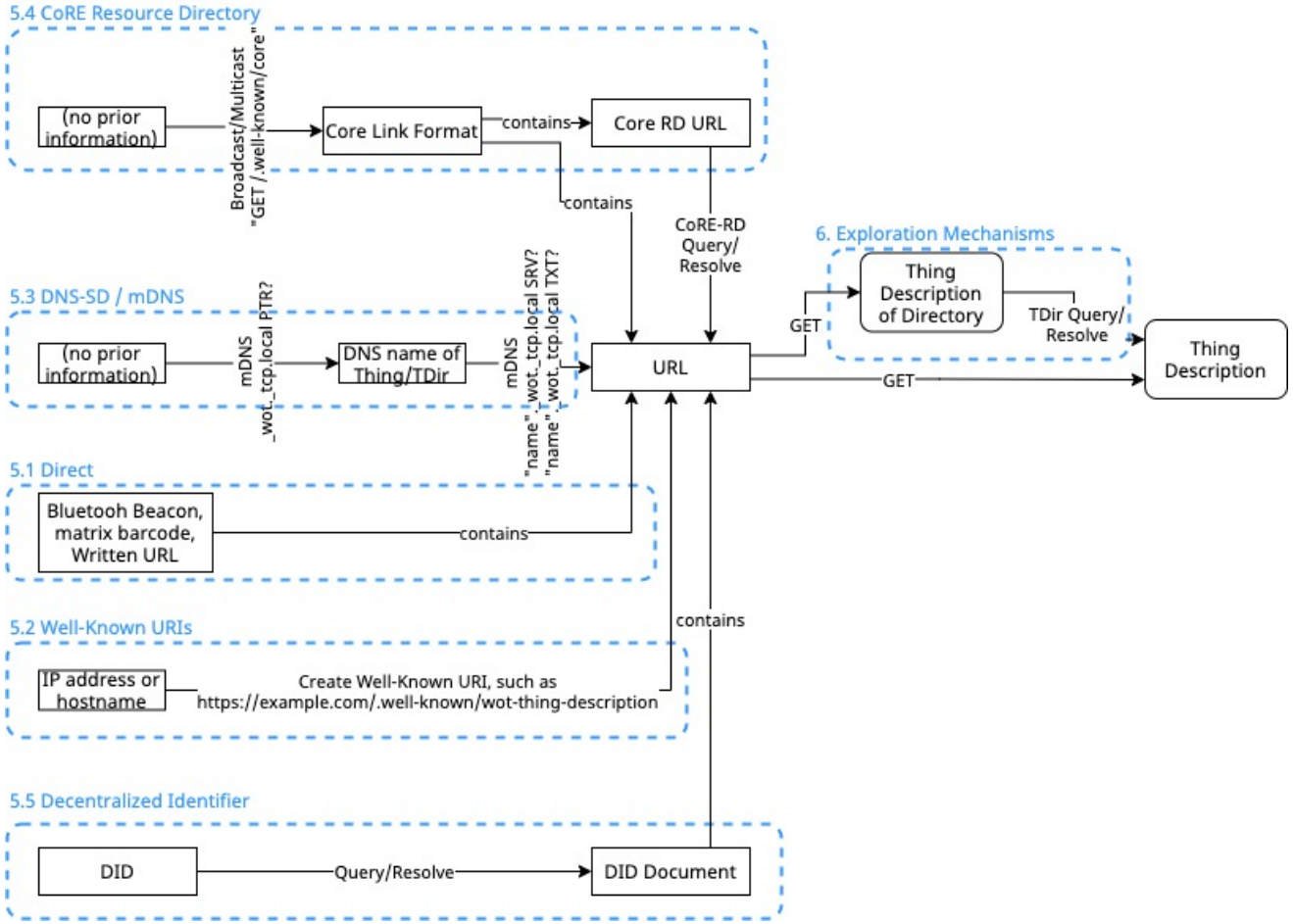
Goal: Obtain TD of interest

- Not limited to local network
- Scalable to many TDs
- Need to preserve privacy
- Phased access:
  1. Introduction: open
  2. Exploration: controlled
- Searchable via JSON Path, XPath, or SPARQL

- Future work:
  - Find "nearby" Things using geospatial data

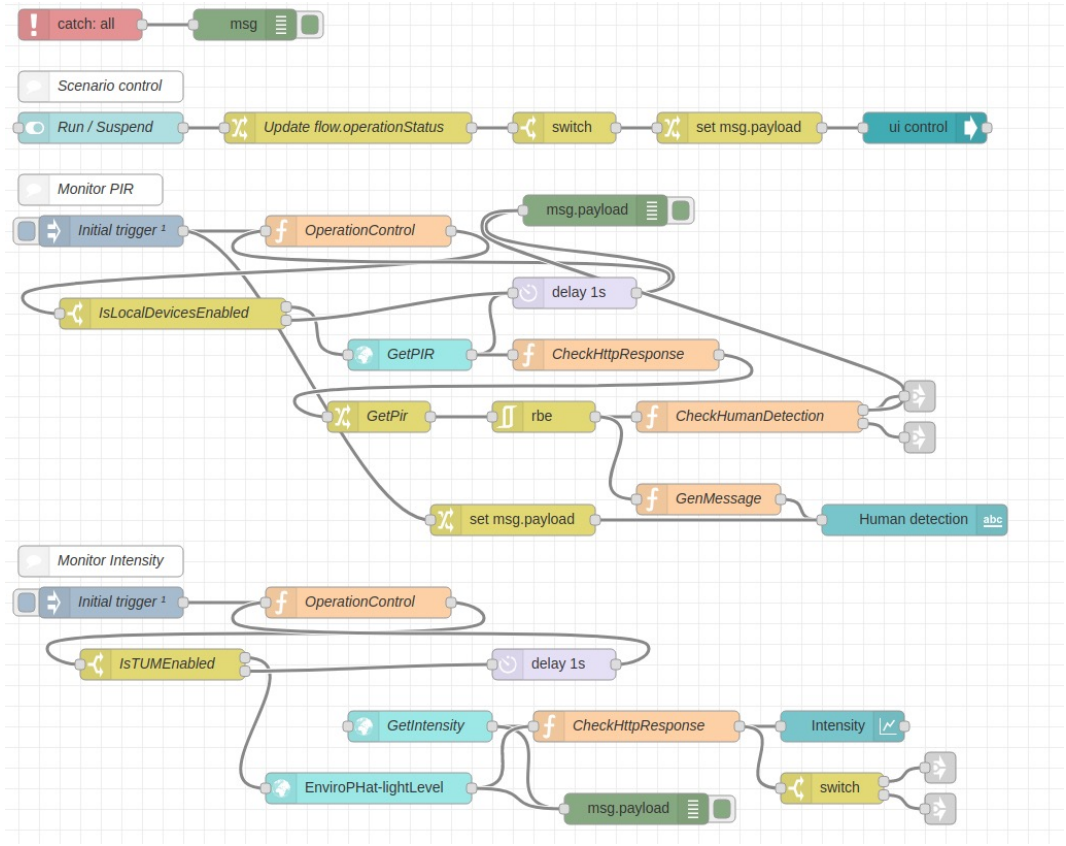
Phase 1: Introduction

Phase 2: Exploration



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); });
```

Use Cases and Requirements

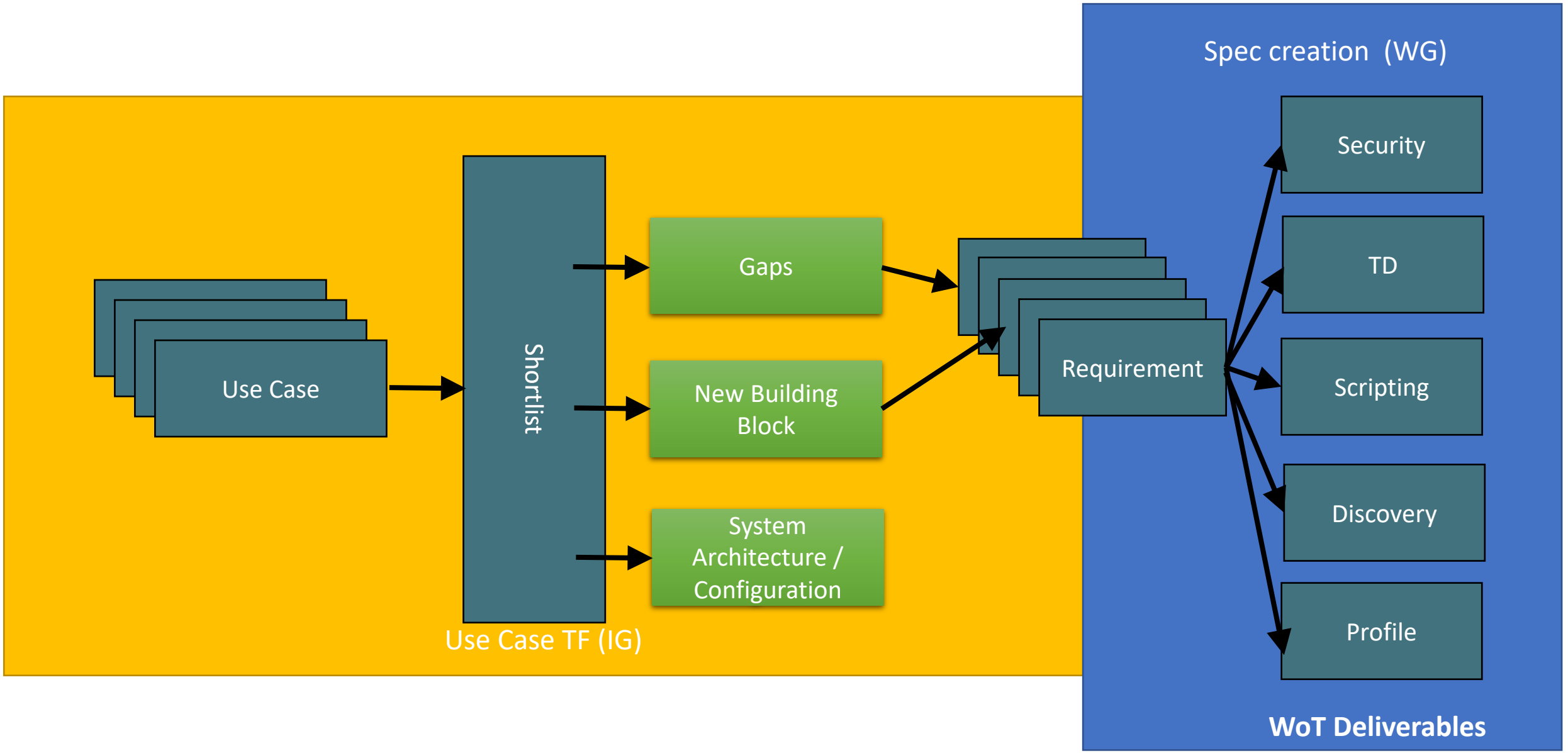
Informative Deliverable: https://github.com/w3c/wot-usecases

Purpose and Process:

- Identify specific use cases
- Identify application domains
  - Collect use cases from other W3C groups
  - Collect use cases from other stakeholders and SDOs
- Identify usage patterns
  - For example, hubs, proxies, automation, etc.
- Identify relevant technologies
  - For example, edge computing, digital twins, etc.

→ Extract common requirements to drive current and future work

Use Cases / Requirements Process



Deliverables

New/Updated Normative Documents:

- 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:

- 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

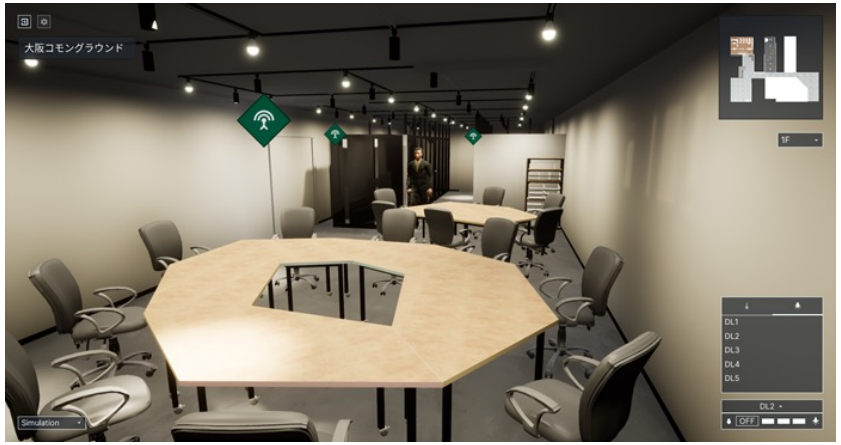
Community Resources:

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

Recent Activity

- Upcoming Plugfest/Testfest: March 14-18
  - https://github.com/w3c/wot-testing/tree/main/events
- New Commercial Usages
  - Takenaka Construction – Smart Building Information Management systems
  - Netzo – IoT dashboards and device management
- Directory Implementations
  - WoT Hive, LogiLab (SPARQL based), Fraunhofer LinkSmart
- IETF Relationships: JSON Path, CoreRD, COSE/JOSE, ASDF
- Under Discussion:
  - Geospatial data, Embedded JSON Signatures
  - New Charters/New Deliverables

Recent Commercial Applications



https://www.takenaka.co.jp/news/2021/05/02/

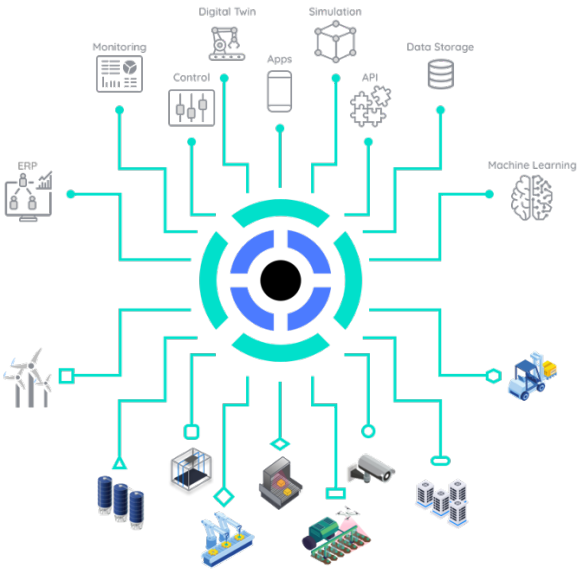
Takenaka Corporation

- CGLL Platform
- BIM applications

https://new.siemens.com/global/en/products/buildings/automation/desigo.html

Siemens

- Desigo CC
- BIM system



https://netzo.io/

Netzo

- IoT Data Hub
- Dashboards

Gaps and Discussion

- GIS Integration
  - Geospatial data and discovery
- Data Management
  - Digital Twins and shadows
  - Event notifications
  - Data management
- Security
  - Key provisioning and onboarding
  - Secure LAN access
  - Proxy services
  - Access control and ad-hoc sharing
  - MUDS
- Accessibility
  - Sensory modality mapping
  - Textual/descriptive interfaces
  - Service location
  - Mobility services
- Advanced Use Cases
  - Transportation
  - Logistics
  - Distributed energy management
  - AR visualization
  - Analytics integration e.g. for health and safety monitoring

Resources and 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