T2TRG: Thing-to-Thing Research Group

IETF 105, July 24, 2019, Montréal

Chairs: Carsten Bormann & Ari Keränen
Note Well

• You may be recorded

• The IPR guidelines of the IETF apply: see http://irtf.org/ipr for details.
Administrivia (I)

- Pink Sheet
- Note-Takers
- Off-site (Jabber, Hangout?)
  - xmpp:t2trg@jabber.ietf.org?join
- Mailing List: t2trg@irtf.org — subscribe at: https://www.ietf.org/mailman/listinfo/t2trg
- Repo: https://github.com/t2trg/2019-ietf105
<table>
<thead>
<tr>
<th>Time</th>
<th>Who</th>
<th>Subject</th>
<th>Docs</th>
</tr>
</thead>
<tbody>
<tr>
<td>13:30</td>
<td>Chairs</td>
<td>Intro, RG status, upcoming meetings and activities</td>
<td>RFC8576, draft-irtf-t2trg-rest-iot</td>
</tr>
<tr>
<td>13:45</td>
<td>Chairs, various</td>
<td>Report from WISHI, Pre-IETF meeting with OMA, Hackathon, and Morning side meeting</td>
<td></td>
</tr>
<tr>
<td>13:55</td>
<td>Michael Koster</td>
<td>Activities on data model convergence; W3C Community Group on Schema extensions for IoT; schema.org update</td>
<td></td>
</tr>
<tr>
<td>14:15</td>
<td>Michael McCool</td>
<td>W3C Web of Things WG/IG update</td>
<td></td>
</tr>
<tr>
<td>14:30</td>
<td>Ivaylo Petrov</td>
<td>YANG Object Universal Parsing Interface</td>
<td>draft-petrov-t2trg-youpi</td>
</tr>
<tr>
<td>14:35</td>
<td>Christian Amsüss</td>
<td>Transports for CoAP: new URI schemes of CoAP protocol negotiation</td>
<td></td>
</tr>
<tr>
<td>14:45</td>
<td>Dirk Kutscher</td>
<td>&quot;Why Edge and IoT will never happen!!1!&quot; (outrageous opinion presentation)</td>
<td></td>
</tr>
<tr>
<td>15:05</td>
<td>Yong-Geun Hong</td>
<td>Problem Statement of IoT integrated with Edge Computing</td>
<td>draft-hong-t2trg-iot-edge-computing</td>
</tr>
<tr>
<td>15:20</td>
<td>Yong-Geun Hong</td>
<td>Edge IoT demo</td>
<td></td>
</tr>
</tbody>
</table>
T2TRG scope & goals

- Open research issues in turning a true "Internet of Things" into reality
- Internet where low-resource nodes ("things", "constrained nodes") can communicate among themselves and with the wider Internet
- Focus on issues with opportunities for IETF standardization
  - Start at the IP adaptation layer
  - End at the application layer with architectures and APIs for communicating and making data and management functions, including security
IRTF and IETF?

CoRE: protocol engineering for RESTful environments

T2TRG: open research issues with IETF potential

LWIG: Informational guidance for implementers
Recent/related activities

- Work on IoT/Semantic Hypermedia Interoperability (WISHI): ~monthly calls and hackathon

- Friday meeting with OMA SpecWorks at IETF 105

- T2TRG work meeting: Wednesday 08:30..09:45. Several small items; focus on secured L3 setup for Things, “Closed Device Groups” (Erik Nordmark)

- Hypermedia/CoRE Applications: Tuesday 15:00..17:00. Mostly discussed new design for the CoRE pub/sub application
Next meetings

- Regular WISHI calls (~ monthly?)
- Virtual meetings with OCF?
- Virtual meetings with OMA SpecWorks (LwM2M & IPSO)?
- Singapore IETF 106 (Nov 16-22)
  - WISHI hackathon Sat/Sun, July 20/21
- Co-locating with academic conferences 2019 & 2020?
Singapore IETF and local collaboration

• Using meetings in specific communities as an opportunity to connect
• IETF106: Singapore
• One obvious point of contact: Singapore “Smart Nation” project
  • They have some requirements on standardization
• Friday T2TRG work meeting?
To deliver city-level data availability for industry and public by 2022, Smart Nation Platform Solutions must be able to demonstrate ability to:

- Standardise, collect and aggregate IOT data at scale
- Guarantee the data comes from authenticated and authorized sources
- Secure the platform end-to-end
- Create immutable records
- Show relevant and sustainable use cases
RG Doc Status

• “State-of-the-Art and Challenges for the IoT Security” published as RFC8576!
“RESTful Design for IoT” (next slide)

Upcoming:

- Edge & IoT (presented later today)
- Secure Bootstrapping for IoT (next slides)
- CoRE apps, collections part from CoRE interfaces
- Layer 3 considerations?
- WISHI notes (see WISHI wiki)
RESTful Design for IoT

• Bunch of small additions / edits done
  • more IoT specifics (commonly constrained servers & dual roles)
  • better and more references
  • server push clarifications & alignment with CoRE dynlink draft
• ToDo: affordances & CoRAL details
• Discovery in IoT? Aligned with CoRE interfaces & RD
Secure Bootstrapping for IoT

- RFC 8576 identifies secure bootstrapping as one of the key challenges for IoT devices

- Plans on future work
  - Document device bootstrapping terminology and relationships: onboarding, commissioning, configuration, setup, initialization
  - Identify common design assumptions, architectural components and underlying protocols that device configuration methods use
  - Investigate the benefits and challenges of EAP for IoT
Work on IoT Semantic/Hypermedia Interoperability (WISHI)

- Two online meetings since IETF104: research agenda & hackathon planning
- Research Agenda topics
  - Modeling data and interaction
  - REST-based hypermedia
  - Connectivity for IoT
  - In-network and edge computing
  - Security
  - Terminology
WISHI hackathon results

• 6th WISHI IETF Hackathon
• ~9 participants (2 remotely)
• Two focus areas
  • IoT Data Model convergence
  • Hypermedia for IoT (and coffee)
IoT Data Model converge

- Using One Data Model (OneDM) Simple Definition Format (SDF) for data and model interchange
  - Improved **automatic conversion** of IPSO/LwM2M models to SDF
  - Improvement suggestion for SDF data types, schema, constraints
  - **Tool** generating SDF schema in CDDL
- Proposed JSON format for CoRAL
Binary data extraction

- **Problem statement** for binary data extraction
- **Playground deployment**
- "YANG Object Universal Parsing Interface": draft-petrov-t2trg-youpi (presented later)
Brewing coffee with hypermedia

- Reference scenario: Carrier-Grade Coffee Machine
  - Discover and describe coffee machine
  - Discover menu options
  - Make coffee selections
  - Brew
- Two open source implementations using CoRAL and CoAP:
  - RIOT OS (running on SAMR21-xpro board)
  - Python (micrurus)
Friday meeting with OMA

• LwM2M tutorial
• Object registry & LwM2M v1.2 requirements
• OMA-IETF document dependencies
  • RD, Dynlink, CoAP over SMS, SenML registry, ...
• Unconference discussions
  • Data model convergence (LwM2M, OneDM, etc.)
  • Role of hypermedia formats (CoRE link format, CoRAL) in LwM2M
  • Access control modeling
• All materials and notes available in the meeting Github
IoT Extensions for schema.org

• Extend schema.org to accommodate IoT semantics
• Develop models for sensors and actuators as a first step
  • With connecting semantics to Features of Interest
• Based on a popular emerging meta-model
  • Properties, Actions, Events => Capabilities
• Community contribution process in development
  • Modeled after schema.org
iotschema Meta-model

• Semantic model for interaction affordances
• Property
  • Readable and optionally writeable state element
• Action
  • A parameterized incoming state change with rich responses
• Event
  • A parameterized outgoing state change
  • Also can be a message describing a happening
  • Can be delivered asynchronously, proactively
iotschema Meta-model

• Capability
  • A set of Properties, Events, and Action definitions that provide common interaction affordances
  • Related to providing a function of limited scope
  • Defined with semantic meaning
  • For example: on/off control, temperature measurement, thermostatic temperature control,
  • Could be larger aggregations, e.g. air conditioner

• Data Types
  • Associate semantic meaning with data constraints
  • For Example, Temperature data, allowed units, number type
iot.schema.org Conceptual Integration with other ontologies

• Feature of Interest concepts and property types to describe location, equipment, or other classifiers

• For example, BrickSchema definitions from Haystack, GENIVI for Automotive FoI

• Quantity and Units constraints can use QUDT concepts and appropriate identifiers

• SSN, SOSA, SAREF concepts can extend a definition

• Definitions and instances may be annotated using RDF
Integration with other Ontologies

Enables Well-Characterized interactions with Physical Entities

- Feature of Interest, O&M
- Situation, Provenance

Iot.schema.org
Definition

- Quantities, Units, Shapes, Property Value Constraint

Software Affordances for measurement and control
Feature of Interest Properties
Status

• Prototype definitions in JSON-LD are online in an experimental namespace
• Used in W3C WoT work for semantic interoperability
• High level interoperability demonstrations using Node-RED
• Forms based submission option in development
• W3C Community Group started for contributions
• Monthly teleconferences
One Data Model

• Emerging activity to drive data model convergence across various SDOs, vendors, and other organizations
• Developing a common definition language that can describe diverse device descriptions
• Not an API description, depends on Protocol Binding to map to network resources
• Using a similar/same meta-model as iotschema
• The language can be used to create iotschema definitions
One Data Model Example (JSON)

```json
{
    "info": {
        "title": "Example file for ODM",
        "version": "20190424",
        "copyright": "Copyright 2019 Example Corp.",
        "license": "http://example.com/license"
    },
    "namespace": {
        "st": "http://example.com/st/#"
    },
    "defaultnamespace": "st",
    "odmObject": {
        "Switch": {
            "odmProperty": {
                "value": {
                    "type": "string",
                    "enum": ["on", "off"]
                }
            },
            "odmAction": {
                "on": {},
                "off": {}
            }
        }
    }
}
```
info {
  title "Example file for ODM Simple JSON Definition Format"
  version "20190424"
  copyright "Copyright 2019 Example Corp. All rights reserved."
  license http://example.com/license
}

namespace {
  st http://example.com/st/#
}

defaultNamespace st

odmObject {
  Switch {
    odmProperty {
      value {
        type string
        enum [on off]
      }
    }
    odmAction {
      on {}
      off {}
    }
  }
}

One Data Model Example
ODM Status

• About 6 months into the activity
• Operating under a set of liaison agreements
• Weekly teleconferences
• Language definition is progressing
• The language is being tested against models from various SDOs and organizations
• Participation from several members of T2TRG
W3C WoT Update
IETF 105 T2TRG
Montreal Canada July 2019
W3C Web of Things

**Goal: Support IoT Interoperability via Open Standards**

- **W3C WoT Interest Group (IG)**
  [https://www.w3.org/2016/07/wot-ig-charter.html](https://www.w3.org/2016/07/wot-ig-charter.html)
  - Started spring 2015
  - ~200 participants
  - Informal work and outreach
  - “PlugFest” validation with running code
  - Exploration of new building blocks
  - “OpenDays” with external speakers
  - Liaisons and collaborations with other organizations and SDOs
  - Second Workshop on Web of Things held 3-5 June 2019 in Munich

- **W3C WoT Working Group (WG)**
  [https://www.w3.org/2016/12/wot-wg-2016.html](https://www.w3.org/2016/12/wot-wg-2016.html)
  - Started end of 2016 (effectively Feb 2017)
  - ~100 participants
  - Normative work on specific deliverables
  - W3C Patent Policy for royalty-free standards
  - Only W3C Members and Invited Experts
  - Architecture and Thing Description were published as Candidate Recommendations on 16 May 2019
  - Notes published on Protocol Bindings, Security, and Scripting API
The index.html for Things

Properties

Events

Actions

The index.html for Things

WoT Architecture

Overarching umbrella with architectural constraints and guidance on how to use and combine building blocks.

WoT Thing Description (TD)

JSON-LD representation format to describe Thing instances with metadata. Uses formal interaction model and domain-specific vocabularies to uniformly describe how to use Things, which enables semantic interoperability.

Security Guidelines

WoT Scripting API

Standardized JavaScript object API for an IoT runtime system similar to the Web browser. Provides an interface between applications and Things to simplify IoT application development and enable portable apps across vendors, devices, edge, and cloud.

WoT Binding Templates

Capture how the formal Interaction Model is mapped to concrete protocol operations (e.g., CoAP) and platform features (e.g., OCF). These templates are re-used by concrete TDs.
**W3C Web of Things – Building Block Approach**

**WoT Architecture**
Overarching umbrella with architectural constraints and guidance on how to use and combine building blocks.

**WoT Thing Description (TD)**
JSON-LD representation format to describe Thing instances with metadata. Uses formal Interaction model and domain-specific vocabularies to uniformly describe how to use Things, which enables semantic interoperability.

**Security Guidelines**

**WoT Scripting API**
Standardized JavaScript object API for an IoT runtime system similar to the Web browser interface between applications and Things to simplify IoT application development and enable portable apps across vendors, devices, edge, and cloud.

**WoT Binding Templates**
Capture how the formal Interaction Model is mapped to concrete protocol operations (e.g., CoAP) and platform features (e.g., OCF). These templates are re-used by concrete TDs.

The `index.html` for Things

Properties
Events
Actions

HTTP
MQTT
CoAP
Published Candidate Recommendations

• **WoT Architecture**
  – Constraints that define the difference between IoT and W3C WoT
  – Definition of Interaction Affordances
  – Definition of Web forms
  – Use cases and requirements
  – Terminology
  – Interplay of W3C WoT building blocks
  – Examples

• **WoT Thing Description (TD)**
  – Information model & representation format for Thing metadata, generic data model, and hypermedia-based interface descriptions
  – Namespace and vocabulary definitions
  – Parsing and serialization rules
  – Extension points
  – Examples
Published Candidate Recommendations

• WoT Architecture
  – Constraints
    ▪ Things must have TD (W3C WoT)
    ▪ Must use hypermedia controls (general WoT)
      – URIs
      – Standard set of methods
      – Media Types
  – Interaction Affordances
    ▪ Metadata of a Thing that shows and describes the possible choices (what) to Consumers, thereby suggesting how Consumers may interact with the Thing

• WoT Thing Description (TD)

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

Door = Thing

Handle = Affordance

What? How?

Open

Pull

Turn
Published WG Notes

- **WoT Security and Privacy Guidelines**
  - Details beyond the security considerations in each specification for a holistic security and privacy configuration of Things
  - Security testing plan

- **WoT Binding Templates**
  - Documentation for how to describe existing IoT ecosystems (e.g., OCF or generic Web) with WoT Thing Description

- **WoT Scripting API**
  - Proposal for a standard API to consume and produce WoT Thing Descriptions
  - Provides interface between applications and network-facing API of IoT devices (cf. Web browser APIs)
  - Documents learnings from the design process
Status and Recent Developments

• Decision to adopt JSON-LD 1.1 proposed features to allow:
  – Default values
  – Object notation (name: value) instead of arrays
  – More similarity to common JSON practices

• Security metadata
  – Focus on HTTPS (Basic Auth, Digest, Tokens, OAuth2)

• Protocol Bindings
  – Focus on HTTP and structured payloads compatible with JSON
  – Support for Events also using subprotocols (e.g., long polling in HTTP)

• Extension Points
  – CoAP(S), MQTT(S), and further security schemes (e.g., ACE)
  – Semantic annotations with custom vocabularies (JSON-LD @context and @type)
W3C WoT Summary

• Counter fragmentation in the IoT
  – Web of Things to Internet of Things is similar to the Web to Internet relation
  – Narrow waist: common interaction model and metadata description
  – Take patterns from the World Wide Web and adapt and apply them to the IoT
    ▪ JSON Schema and Linked Data
    ▪ URIs and Media Types
    ▪ JavaScript runtime

• By describing and complementing
  – Not competing with existing IoT standards, as not prescribing a full-stack solution
  – Instead, describes existing solutions so they can work with each other (interoperate)
  – W3C WoT defines common building blocks to enable semantic interoperability
    ▪ WoT Thing Description (TD)
    ▪ WoT Binding Templates
    ▪ WoT Scripting API
W3C WoT Resources

- **W3C WoT Wiki**
  - https://www.w3.org/WoT/IG/wiki
    (IG/WG organizational information)

- **W3C WoT Interest Group**
  - https://www.w3.org/2016/07/wot-ig-charter.html
    (charter)
  - https://lists.w3.org/Archives/Public/public-wot-ig/
    (mailing list)
  - https://github.com/w3c/wot
    (technical proposals)

- **W3C WoT Working Group**
  - https://www.w3.org/2016/12/wot-wg-2016.html
    (charter)
  - https://www.w3.org/WoT/WG/
    (dashboard)

- **W3C WoT Candidate Recommendations**
  - https://www.w3.org/TR/wot-architecture/
  - https://www.w3.org/TR/wot-thing-description/

- **W3C WoT Working Drafts / Group Notes**
  - https://www.w3.org/TR/wot-binding-templates/
  - https://www.w3.org/TR/wot-scripting-api/
  - https://www.w3.org/TR/wot-security/

- **W3C WoT Editors’ Drafts and Issue Tracker**
  - https://github.com/w3c/wot-architecture/
  - https://github.com/w3c/wot-thing-description/
  - https://github.com/w3c/wot-binding-templates/
  - https://github.com/w3c/wot-scripting-api/
  - https://github.com/w3c/wot-security/

- **Reference Implementation: node-wot**
  - https://github.com/eclipse/thingweb.node-wot
Contacts

https://www.w3.org/WoT/WG/

Dr. Michael McCool  
Principal Engineer  
Intel  
Technology Pathfinding  
michael.mccool@intel.com

Dr. Matthias Kovatsch  
Principal Researcher  
Huawei Technologies  
Applied Network Technology Lab  
matthias.kovatsch@huawei.com
"Transport Design Team"

Transports for CoAP
new URI schemes of CoAP, protocol negotiation

Bill Silverajan, Klaus Hartke, Ines Robles, Christian Amsüss

2019-07-24
“Mini-charter”

G1 Define CoAP over SMS (work in CoRE)

G2 Single scheme for all transports (avoiding URI aliasing)

G3 Announcing the active transports (allow transport switchovers)
“Mini-charter”

**G1** Define CoAP over SMS (work in CoRE)
- So we’re ready for other non-IP ones (NB-IoT, slipmux, …)
- Starting from old `coap-over-sms` and OMA LwM2M input

**G2** Single scheme for all transports (avoiding URI aliasing)

**G3** Announcing the active transports (allow transport switchovers)
“Mini-charter”

G1 Define CoAP over SMS (work in CoRE)
   ▶ So we’re ready for other non-IP ones (NB-IoT, slinemux, ...)
   ▶ Starting from old coap-over-sms and OMA LwM2M input

G2 Single scheme for all transports (avoiding URI aliasing)
   ▶ input from HTTP’s Alt-Svc

G3 Announcing the active transports (allow transport switchovers)
“Mini-charter”

G1 Define CoAP over SMS (work in CoRE)
   ▶ So we’re ready for other non-IP ones (NB-IoT, slipmux, ...)  
   ▶ Starting from old coap-over-sms and OMA LwM2M input

G2 Single scheme for all transports (avoiding URI aliasing)
   ▶ input from HTTP’s Alt-Svc

G3 Announcing the active transports (allow transport switchovers)
   ▶ Starting from protocol-negotiation
Participation appreciated

https://github.com/t2trg/transports
Problem statement

- Discussed during the WISHI hackathon
- LPWAN and other very constrained networks use proprietary binary formats (including Modbus)
- Other systems can not easily interoperate with those
- Needs a format to express their binary payloads and be able to reformat it as CBOR/JSON/JSON-LD/XML/something else
What it is

... typedef battery-level {
   type decimal64 {
      fraction-digits 2;
      range "3 .. 4.2";
   }
   description "CHANGEME";
   units "<units uri>";
   youpi:units-subject "<item id>";
}
...

leaf battery {
   type battery-level;
   youpi:position "8..11 | 7";
   youpi:multiplier "0.05";
   youpi:offset "54";
}

choice data {
   case _temp {
      container button-data {
         leaf temp {
            type uint8;
            youpi:position "relative 24..29";
            youpi:multiplier "2";
            youpi:offset "54";
            youpi:multiplier "3";
         }
      }
   }
   ...
   youpi:condition "mode";
}
Steps forward

- Check interest
- Try to write models for different specific use cases in order to make sure every important case is supported
- Take it from there
Questions and answers

Thank you!
Why Edge Computing for IoT Will Never Happen

Dirk Kutscher
University of Applied Sciences Emden/Leer
IoT
Bingo!

IoT

Edge Computing
IoT

- Industrial IoT
- Home networks
- Smart City
- Agriculture
- Automotive
Example: Industrial IoT

Enterprise/public cloud realm

Factory/enterprise DC realm

Factory Floor Realm

Deterministic Networking Realm

Data analytics, archival

Cloudified control apps (virtual PLC etc.)

Data exchange & control (OPC UA, DDS)

Ethernet Time Sensitive Networking (TSN)

<table>
<thead>
<tr>
<th>Standard</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>802.1ASrev</td>
<td>Timing &amp; Synchronization</td>
</tr>
<tr>
<td>802.1Qbv</td>
<td>Enhancements for Scheduled Traffic (Timed Gates for Egress Queues)</td>
</tr>
<tr>
<td>802.1Qbu</td>
<td>Frame Preemption</td>
</tr>
<tr>
<td>802.1Qsa</td>
<td>Path Control and Reservation</td>
</tr>
<tr>
<td>802.1Qtc</td>
<td>Central Configuration Management</td>
</tr>
<tr>
<td>802.1Qti</td>
<td>Per-Stream Time-based Ingress Filtering and Policing</td>
</tr>
<tr>
<td>802.1CB</td>
<td>Redundancy Frame Replication &amp; Elimination</td>
</tr>
</tbody>
</table>
Example:
IoT Data Processing
Technical Criteria of Interest

• What are the interaction models?
  • I.e., stateless functions vs. server/actor model

• What are the objects of computation?
  • I.e., packets/flows, Application Data Units etc.

• What are the programs?
  • (Mobile) code

• What is the security & trust model?
  • Postponing that discussion not acceptable
Edge Computing — Too Broad

Hannu Flinck: https://datatracker.ietf.org/meeting/98/materials/slides-98-nfvrg-sessb-12-multi-access-edge-computing-mec-applications-00
Do You Mean…

- Virtualized gateway platforms?
- Cloud-to-edge continuum?
- Compute offload in constrained networks?
- Distributed computing, stream processing?

Often, Edge Computing seems to refer to Cloud Computing with additional compute pods outside a data center.
Is Edge Computing the Best Term?

Cloud spreads to the edge…
Is Edge Computing the Best Term?

Centralization

Rent-seeking business models

Dependencies

Cloud spreads to the edge…
Restart Discussion: Computing with Things

- Computing for/with things
  - Without enforcing dependencies on centralized communication/computing, and security infrastructure
- CoAP mindset
  - Building blocks that can be used to realize different application/business requirements
  - Without solving all the problems in the world…
Potentially Interesting Directions

- Decentralized, secure Computing with Things
  - Connect things in local network
  - Establish trust
  - Offload computation
  - (Does not exclude talking to cloud, but that should not be the mental model to start with)
From Overlays…
From Overlays...

- Circuit-like connectivity
- Limited visibility into network
- Different namespaces
- Need additional infrastructure find things, compute platforms, functions
- DNS, discovery
To Computing in the Network with Joint Resource Optimization

- Do not require fixed locations of data and computation
- Lay out processing graphs flexibly
- Sometimes we can move functions (close to big data assets)
- At other times we gradually move data where it is needed (e.g., where specific computations run)
- Conditions may change dynamically and constantly: network to adapt to application requirements, network conditions etc.
- Avoiding dependencies on orchestrators
Suggested Environment: Computing with Constrained Things

• Function offloading (power saving, load management)

• Triggered execution, reactive programming, IFTTT

• Custodial transfer (data offloading)

• Data processing pipelines
Summary

• Let’s not boil the ocean and survey all possible combinations of IoT and Edge Computing

• There are many forums, alliances etc. that do something in that space — where can we make a dent (and do good research)?

• Suggesting application-driven technology development for selected specific environments (e.g., constrained networks)

• Important to dive deeper than just to the business case level

• Interactions models, computation models etc.

• Pillars: decentralized, leight-weight, joint optimisation of networking and computing, object security

• T2TRG activity could dove-tail with COIN work, but focus on these pillars
Problem Statement of IoT integrated with Edge Computing (draft-hong-t2trg-iot-edge-computing-00)

IETF105 T2TRG meeting in Montreal

J. Hong, **Y-G. Hong**, X. de Foy, M. Kovatsch, E. Schooler and D. Kutscher
Contents

• History and major updates on draft

• IoT Edge computing demo show
  • To support the draft
History of the draft

• IETF 103
  • Presented first in T2TRG side meeting
  • draft-hong-iot-edge-computing-01
  • Showed two demo videos as use cases of IoT Edge computing
    • Smart constructions providing a monitoring service of construction site
    • Real-time control monitoring system by Rotary Inverted Pendulum system

• IETF 104
  • Presented in Pre IETF 104 work meeting
  • draft-hong-iot-edge-computing-02
Major Updates

• Changed the filename to specify it under T2TRG
  • draft-hong-t2trg-iot-edge-computing-00
    • It was draft-hong-iot-edge-computing-02

• Integrated with *Survey and gap analysis*
  • It was presented and discussed at IETF100 T2TRG

• New authors are added
  • Xavier de Foy (InterDigital Communications)
  • Matthias Kovatsch (Huawei Technologies Duesseldorf GmbH)
  • Eve Schooler (Intel)
  • Dirk Kutscher (University of Applied Sciences Emden/Leer)
Changes of each chapters (1/3)

3. Background ........................................ 3
   3.1. Internet of Things (IoT) .................. 3
   3.2. IoT with Cloud computing .............. 4
   3.3. IoT Environmental changes ............. 4
4. New challenges of IoT ......................... 4
   4.1. Strict Latency .............................. 5
   4.2. Constrained Network Bandwidth ......... 5
   4.3. Constrained Devices ..................... 5
   4.4. Uninterrupted Services with Intermittent Connectivity to the Cloud ............ 5
   4.5. Privacy and Security .................... 79

<draft-hong-iot-edge-computing-02>

<draft-hong-t2trg-iot-edge-computing-00>
Changes of each chapters (2/3)

<draft-hong-iot-edge-computing-02>

5. IoT integrated with Edge Computing..........
  5.1. IoT Data in Edge Computing..........
    5.1.1. Data Storage..........
    5.1.2. Data Processing..........
    5.1.3. Data Analyzing..........
  5.2. IoT Device Management in Edge Computing..
  5.3. Edge Computing in IoT..........
  5. Architecture of IoT integrated with Edge Computing
  7. Use Cases of Edge Computing in IoT..........
    7.1. Smart Constructions..........
    7.2. Smart Grid..........
    7.3. Smart Water System..........
    7.4. Smart Buildings..........
    7.5. Smart Cities..........
    7.6. Connected Vehicles..........

<draft-hong-t2trg-iot-edge-computing-00>

5. IoT integrated with Edge Computing..........
  5.1. IoT Data in Edge Computing..........
    5.1.1. Data Storage..........
    5.1.2. Data Processing..........
    5.1.3. Data Analyzing..........
  5.2. IoT Device Management in Edge Computing..
  6. Architecture of IoT integrated with Edge Computing
  7. State-of-the-art of IoT Edge Computing
    7.1. Common aspects of IoT edge computing service platforms
    7.2. Use Cases of IoT Edge Computing
Changes of each chapters (3/3)

Appendix A. Overview of the IoT Edge Computing ........................................................................ 17
A.1. Open Source Projects ........................................................................................................ 17
    A.1.1. Gateway/CPE Platforms ............................................................................................. 17
    A.1.2. Edge Cloud Management Platforms ........................................................................ 18
    A.1.3. Related Projects ....................................................................................................... 19
A.2. Products .............................................................................................................................. 19
    A.2.1. IoT Gateways ............................................................................................................ 19
    A.2.2. Edge Cloud Platforms ............................................................................................. 20
A.3. Standards Initiatives .......................................................................................................... 20
    A.3.1. ETSI Multi-access Edge Computing ........................................................................ 20
    A.3.2. Edge Computing Support in 3GPP .......................................................................... 21
    A.3.3. OpenFog Consortium ............................................................................................... 22
    A.3.4. Related Standards .................................................................................................... 22
A.4. Research Projects ............................................................................................................. 22
    A.4.1. Named Function Networking .................................................................................... 22
    A.4.2. 5G-CORAL ................................................................................................................ 23
    A.4.3. FLAME ...................................................................................................................... 23
Gateway-based architecture of IoT Edge Computing

• This is one particular way of doing Edge computing

• Provides
  • downside connectivity to IoT sensors and devices (southbound connectivity)
  • upside connectivity to cloud networks (northbound connectivity)
  • function of data storage
  • computing function such as data processing, data analyzing, and intelligence
Next revision & Direction

• Provides the different Edge computing approaches
  • edge cloud, edge gateway, distributed edge nodes, device-embedded edge nodes, etc.

• T2TRG adoption?
IoT Edge computing demo

- ETRI implementation -
Object of demonstration

• Show an implementation of Edge computing based on open source EdgeX

• Provide a mapping between implementation & architecture in the draft

• T2TRG adoption support
Service Scenario

IoT Edge Computing System

- Arduino R3
- IoT
- Edge Computing System
- Sensing Box
- Object Detector
- Mongo DB
- Analytics
- Sensors (sound level meter)
- LTE modem
- Object Detection
- Raspberry pi 3 B+, Movidius NCS2

Sensor data
Control data

IETF105 T2TRG meeting
Example of Edge computing function: Intelligence - Preprocessing, Prediction, Analyze & Control

Cloud Networks

Sensor Data

Pre-Processing

Training

Model

Saved model (.pb)

Edge Networks

Sensor Data

Pre-Processing

Inference

Model Import

Saved model (.pb)
Service Scenario – Normal vs. Abnormal

**Normal Situation**

- **Pre-processing**
- **Model**
- **inference**

**Abnormal Situation**

- **Pre-processing**
- **Model**
- **inference**

**Thing 1 (Sensors)**

**Thing 2 (Edge System)**

**Thing 3 (Actuator)**
Testbed Configuration

Sound level meter (center 322)  Arduino R3  Edge Laptop  Web-cam  Movidius NCS2  Raspberry pi 3 B+

2019-07-24
Software Configuration (based on EdgeX)

1) Connect Device
   - Sound Device

2) Collect Data
   - MQTT-DS
   - CORE SERVICES
   - KAFKA-ES
   - KAFKA Broker
     - Deep Learning
     - Preprocessing
     - Inference

3) Preprocess Data & Prediction
   - Analyze_control-ES

4) Analyze & Control Device
   - Object-DS
   - CORE SERVICES
   - Object Detector

5) Actuate Device

IETF105 T2TRG meeting
2019-07-24
Screenshot of each process

AI Platform
AI Platform Web-toolkit
MQTT-DS
Analyzing-control-ES + Object-detection-DS
Sound Sensor
Monitoring using Object Detection

IETF105 T2TRG meeting
1) Connect Device

sound level meter (center 322)

- Arduino R3
- Linux
- MQTT
- EMQ MQTT Broker
- EdgeX

• MQTT Producer programing
• MQTT data processing
2) Collect Data

```cpp
## SENSOR$FWVER:1.0
:GAS:0.0,
:SOUND: 65.32\n
{ ID: adfb32432dbf3
  Name: sound-meter-01
  Payload: data
}

{ Topic: DataTopic
  Origin: 124d56fad
  Name: Value
  Value: 65.34
}
```

Get:
/api/v1/reading

Subscribe:
/DataTopic, get value

Publish:
/DataTopic, value

---

IETF105 T2TRG meeting

2019-07-24
3) Preprocess Data & Prediction
4) Analyze & Control Device

Analyze

Control Device

Visualize
5) Actuate Device

Object-DS

Object-Detector

Raspberry pi 3 B+

Movidius NCS2

Logitech camera

IETF105 T2TRG meeting

2019-07-24
Gateway-based architecture of IoT Edge computing
How our implementation is related to the draft

1) Connect Device
2) Collect Data
3) Preprocess Data & Prediction
4) Analyze & Control Device
5) Actuate Device

Sound Device
Object Detector
MQTT-DS
KAFKA-ES
KAFKA Broker
Deep Learning
Preprocessing
Inference
Analyze_control-ES
Object-DS
Core Services

Cloud networks
Edge gateway function (Northbound)
Edge networking function (Northbound)

2019-07-24
Thanks!!

Questions & Comments