T2TRG: Thing-to-Thing Research Group

IETF 102
July 19, 2018, Montréal, CA

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/2018-ietf102**

Agenda

Time	Who	Subject	Docs
15:50	Chairs	Intro, RG Status	draft-irtf-t2trg-iot-seccons-15 draft-irtf-t2trg-rest-iot-01
16:00	Michael Koster / Chairs	Report from WISHI and Hackathon	
	Michael Koster	iot.schema.org update	
	Matthias Kovatsch	W3C Update	
16:40		Next steps in security	draft-garciamorchon-t2trg-automated-iot-security
17:40	Chairs	Wrap-up	
17:50		Meeting ends	

Next Steps in Security

- Oscar Garcia-Morchon: Automated IoT Security
- Mohit Sethi: Enabling Network Access for IoT devices from the Cloud
- René Struik: Next Steps in Security
- Dirk Kutscher: Decentralized Trust for IoT and In-Network-Computing
- Carsten Bormann: IoT Security Semantics and Semantics Security

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

Recent activities

- Work on IoT/Semantic Hypermedia Interoperability (WISHI): bi/tri-weekly calls and hackathon
- “State-of-the-Art and Challenges for the IoT Security” ready for publication
- Joint WebEx sessions with OCF on CoRE technologies: CoRE Resource Directory, Dynamic Linking, REST conventions, Object Security
- Starting to kick-off joint work with OMA SpecWorks

Identified Research Topics

- Good executable models that enable extracting information from **byte strings** and upgrading to data model level of JSON/CBOR

- Generate one worked example: Semantics of state of and operations on a **light** (seriously, this is not as trivial as it sounds)

- Looking at various description techniques and models from other SDOs; how do they handle **protocol evolution** and how can improve

Next meetings

- Regular WISHI calls (~ monthly)
- Virtual meetings, F2F? with OCF
- Virtual meetings with OMA SpecWorks (LwM2M & IPSO)
- Bangkok IETF 103
  - IoT Edge Computing session?
  - WISHI hackathon?
- Co-locating with academic conferences 2019?

Edge and In-Network computing

- Multiple RGs with relevant activities:
  T2TRG, ICNRG, DINRG, PEARG
- E.g.,
  - recent submission:
    draft-hong-iot-edge-computing
  - IETF 100 Edge Computing T2TRG session
- Joint meeting at IETF 103 (Bangkok)?

RG Doc Status

- “State-of-the-Art and Challenges for the IoT Security” ready
- “RESTful Design for IoT” (next slide)
- Upcoming:
  - Document(s) to be shaped from CoRAL and CoRE Apps?
  - Inter-network Coexistence in IoT?

RESTful Design for IoT

- Hypermedia guidance included in -01
- More IoT specifics throughout the draft
  - Role of REST constraints for IoT
  - System characteristic: REST used for scaling down & need to evolve without simultaneous updates
- Terminology updates: dereference & -able URI. Nondeferencable URI example of dev:urn

Next steps with RESTful IoT

- Author review round for internal consistency.
  See the Github draft for latest.
- Revive, summarize, and reference CoRE Apps
- Submit new version for broader review (e.g., microservices community)

Not a RG document: draft-sarikaya-t2trg-sbootstrapping

- Survey of security bootstrapping methods
  - Originally a 6LoWPAN document
  - Further developed in T2TRG
- Not much feedback
- Do we have the energy (and the interest) to evolve that as a RG document?

Agenda

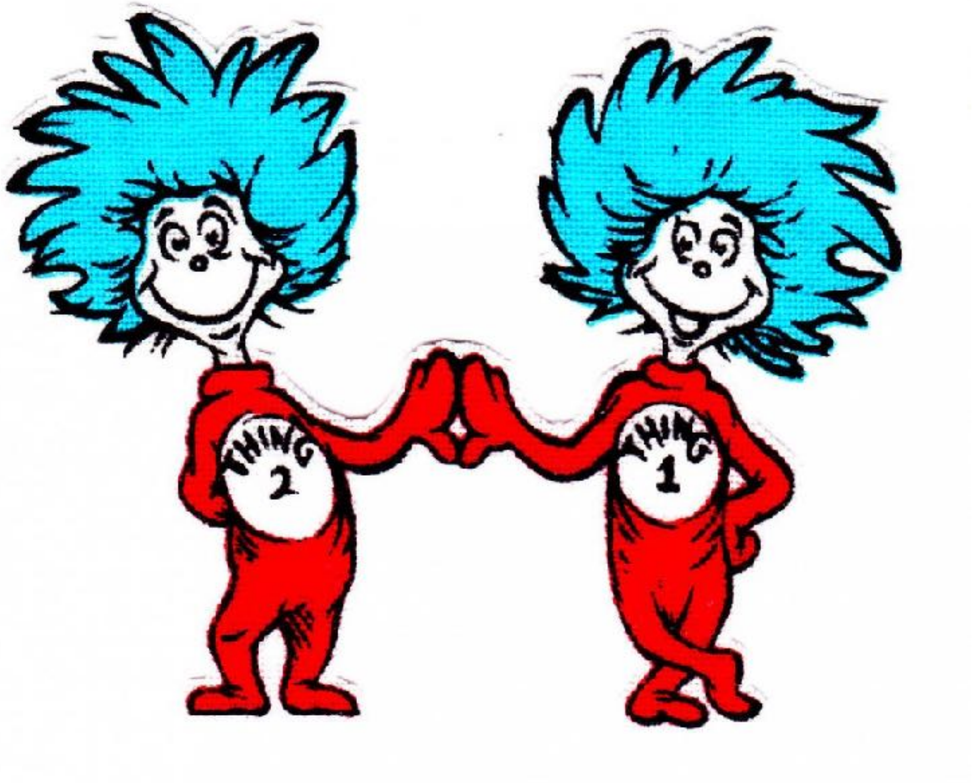
Time	Who	Subject	Docs
15:50	Chairs	Intro, RG Status	draft-irtf-t2trg-iot-seccons-15 draft-irtf-t2trg-rest-iot-01
16:00	Michael Koster / Chairs	Report from WISHI and Hackathon	
	Michael Koster	iot.schema.org update	
	Matthias Kovatsch	W3C Update	
16:40		Next steps in security	draft-garciamorchon-t2trg-automated-iot-security
17:40	Chairs	Wrap-up	
17:50		Meeting ends	

WISHI
IETF 102 Hackathon
iot.schema.org

T2TRG - IETF 102

Michael Koster

Thing 2 Thing…



WISHI

- Work on IoT Semantic and Hypermedia Interoperability
- Bi-weekly teleconferences held between IETF meetings
- Semantic Interoperability hands-on testing and breakout sessions at IETF Hackathons (100, 101, 102)

WISHI Teleconference Topics

- W3C WoT Thing Description (Matthias Kovatsch)
- Processing models for semantic data
- Terminology for layers (Carsten Bormann)
- How to integrate IoT with Energy (Bruce Nordman)
- Impact of JSON LD 1.1 work on Thing Descriptions
- W3C plugfest and WISHI (Matthias Kovatsch)
- WISHI hackathon planning
- Using iot.schema.org with IPSO/LwM2M models

Other SDOs

- Work with devices from IoT ecosystems and SDOs, and tools/specifications from other organizations
  - OCF
  - OMA LWM2M
  - GENIVI VSS (Automotive IVI)

  - W3C Web of Things
  - iot.schema.org

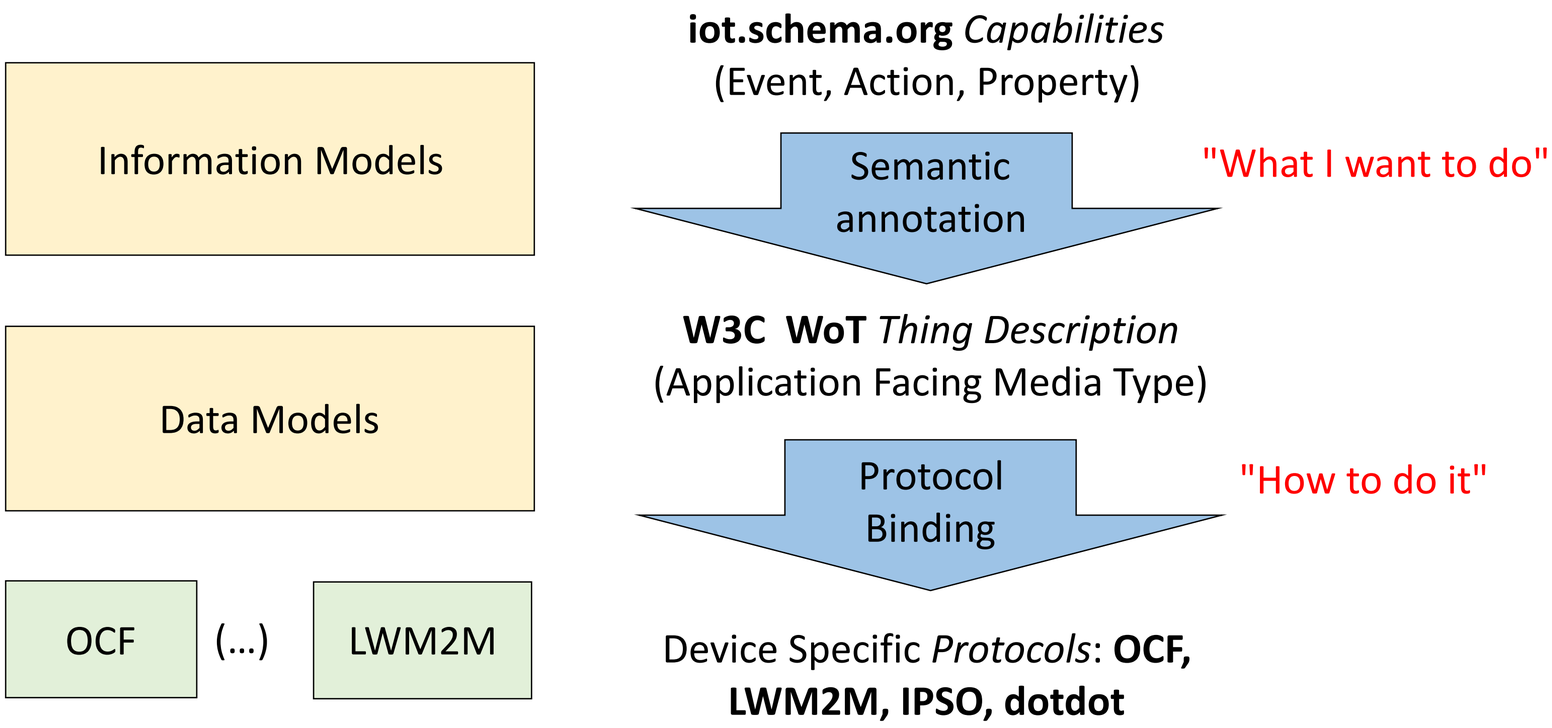
WoT Thing Description

- TD is a file format and mediatype of RDF
- Describes abstract Interactions with things
  - Read temperature
  - Lock the door
  - Change the brightness of a light
- Binds to concrete instances that implement the interactions
  - Defines payload structure
  - Defines data characteristics; type, range, units
  - Transfer layer instructions including URI, methods, options
- Applications use abstract interactions to decouple from the underlying implementation
- The WoT implementation can automatically adapt to the specifics of the device protocol and data formats

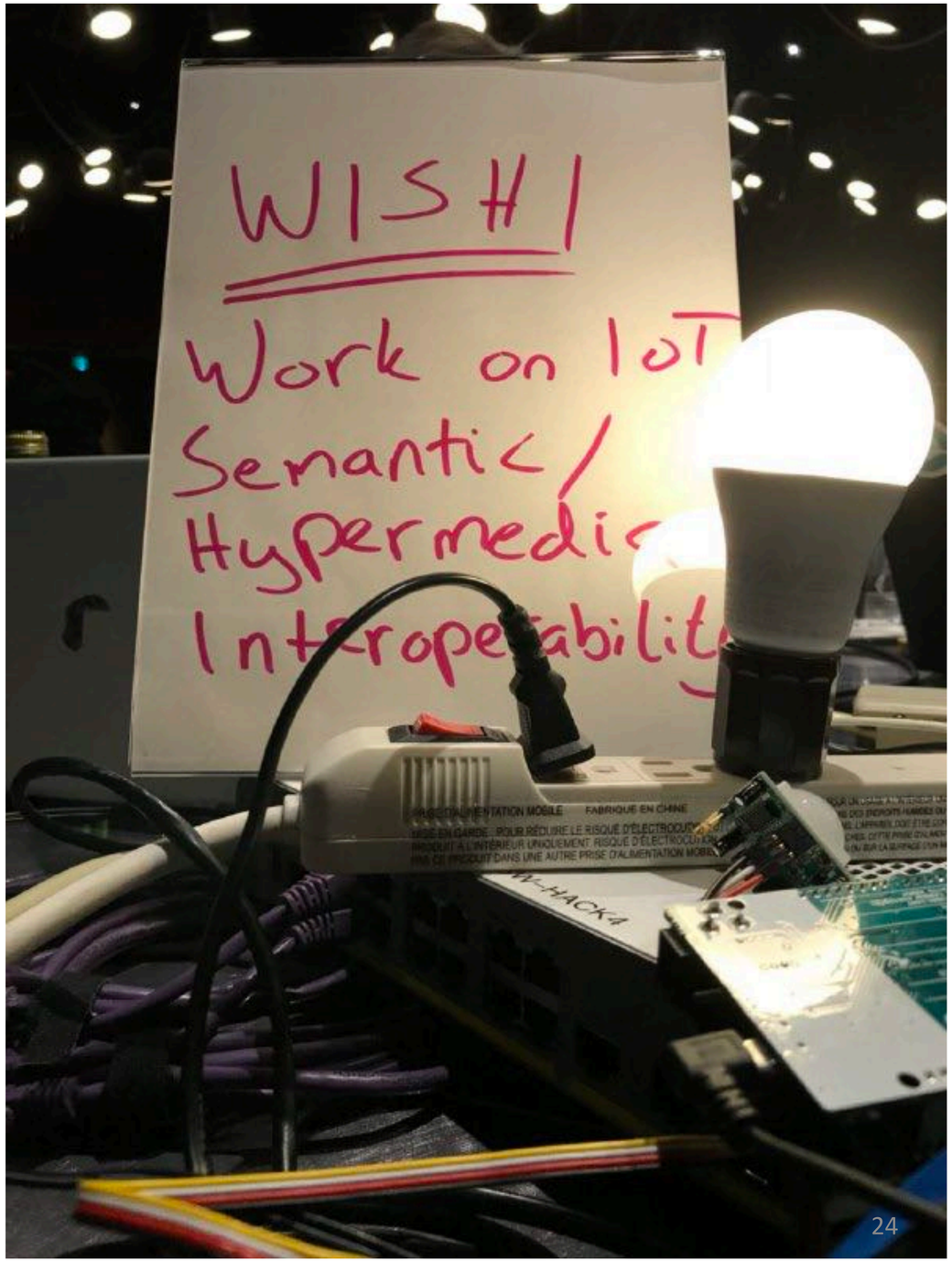
iot.schema.org

- External vocabulary for Semantic Annotation of Thing Description instances
- Semantic Categories to annotate systems at different layers
- High level capabilities with control plane (interaction) and data plane (data item) annotations

Layered Scope in Data Models and Information Models



Hackathon



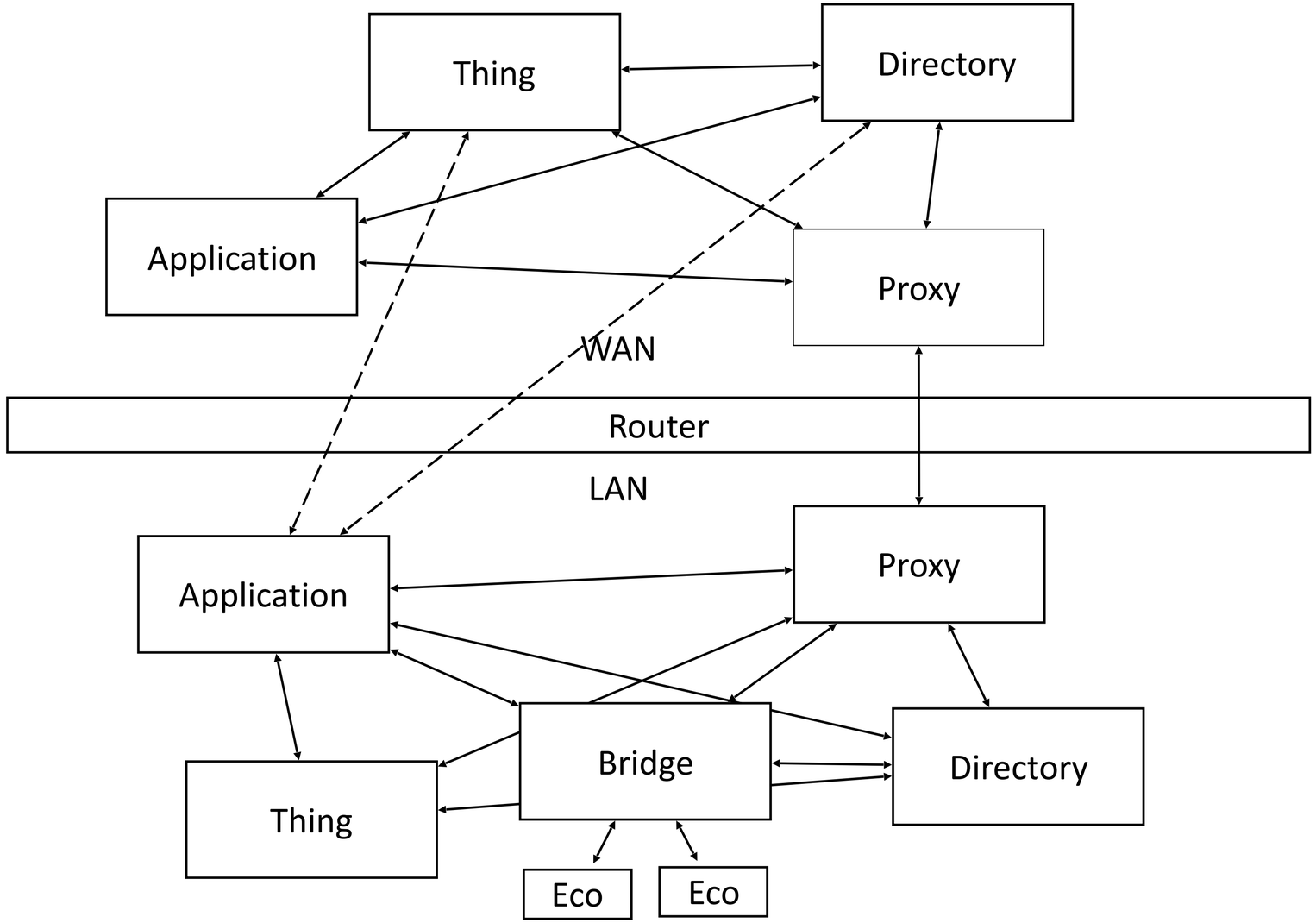
IETF 102 Hackathon

- Overall Goals
- Technical Components
- Projects
- Learning
- Results

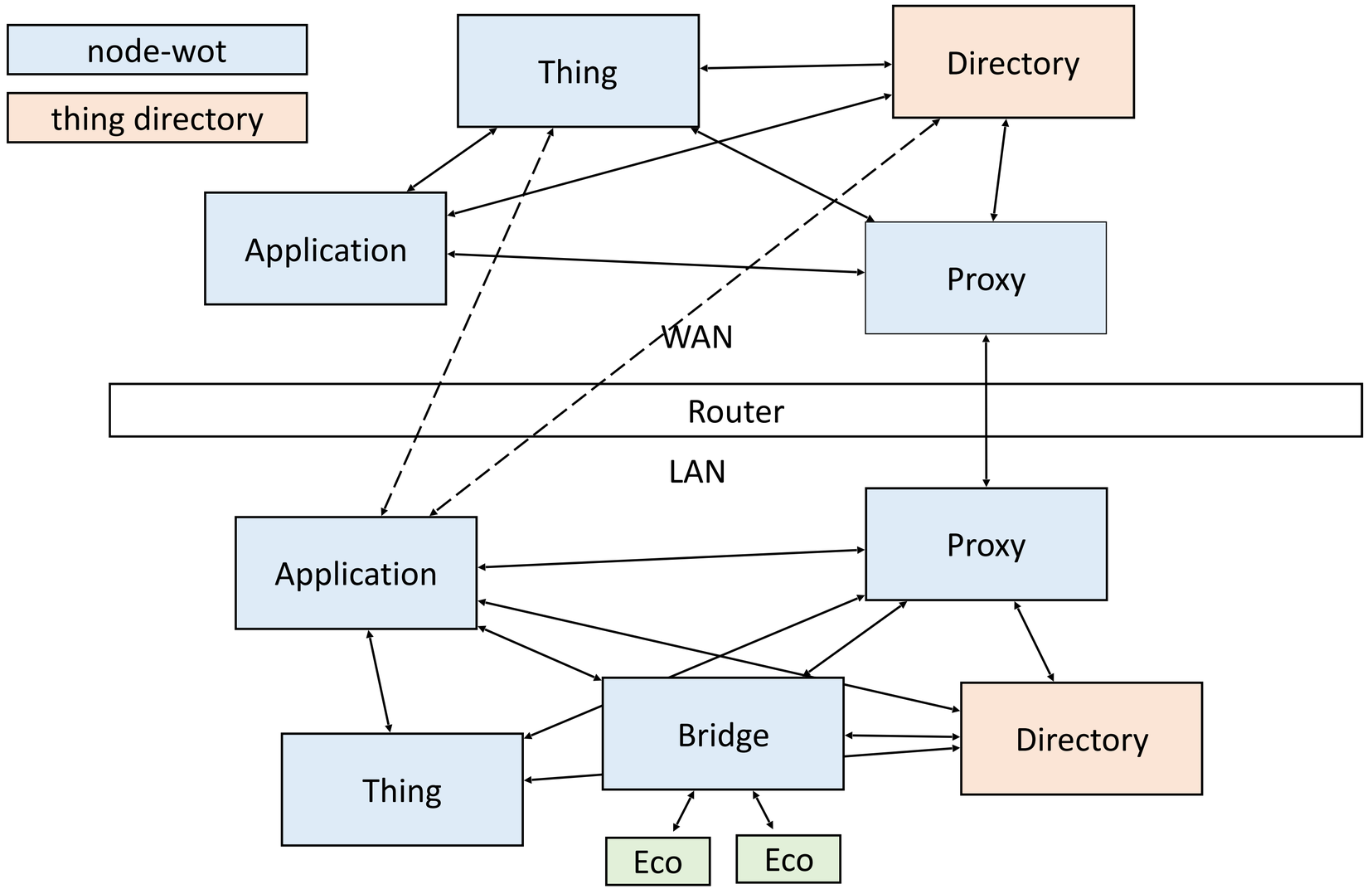
WISHI Hackathon Objectives

- Test Hypermedia and Semantic Interoperability mechanisms in a hands-on environment, with hardware components
- Interoperate across teams/contributors
- Learn about gaps in existing systems and new requirements
- Test extensions and new patterns
- Have breakout discussions to explore issues

System Architecture and Roles



Web of Things Components



Technical Components (1)

- Mediatypes
  - CoRE Link-Format and Web Linking (RFC6690, RFC8288)
  - WoT Thing Description
  - OMA LWM2M
  - SenML
  - JSON
- Protocols
  - HTTP
  - CoAP
  - MQTT
  - DNS-SD

Technical Components (2)

- Software Components
  - Thingweb - node-wot
  - Thingweb - Thing Directory
  - CoRE Resource Directory
  - Node-RED
- Some Bridged Ecosystems
  - OCF
  - LWM2M
  - IKEA Lighting
  - Philips Hue

Projects

- IPSO/LWM2M mapping using WoT Thing Description and iot.schema.org
- OCF mapping using WoT Thing Description and iot.schema.org
- RD Implementation
- W3C Wot Protocol Bindings to CoAP+DTLS devices
- Semantic wrapper for W3C WoT Scripting API
- DNSSD

Some Results

- Breakout discussion on high level work items/areas
- Demonstrated interoperation between generic clients and diverse devices
- Closed 44 issues with node-wot implementation and moved to Eclipse Foundation
- Got RD implementation up to speed and ready to integrate Thing Directory functionality
- Demonstrated automatic interaction with diverse CoAP+DTLS servers
- Report in progress

Breakout Topics

- How we attach models to existing instances of descriptive data / metadata
- Shared models across device ecosystems
- High level semantic API
- Mapping TD to Link-Format
- Discovery use cases and scenarios
- How and where is Semantic Interoperability used?

Learning

- We have a fairly complete stack and tool set to get started – now we should build out
- We should think more about high level applications and test cases for interoperability
- We might think about modeling internal behavior
- Setup time is still a big issue, taking most of the first day
- We should prioritize a way to conduct distributed testing and enable a virtual LAN
- We need alternate implementations of critical functions

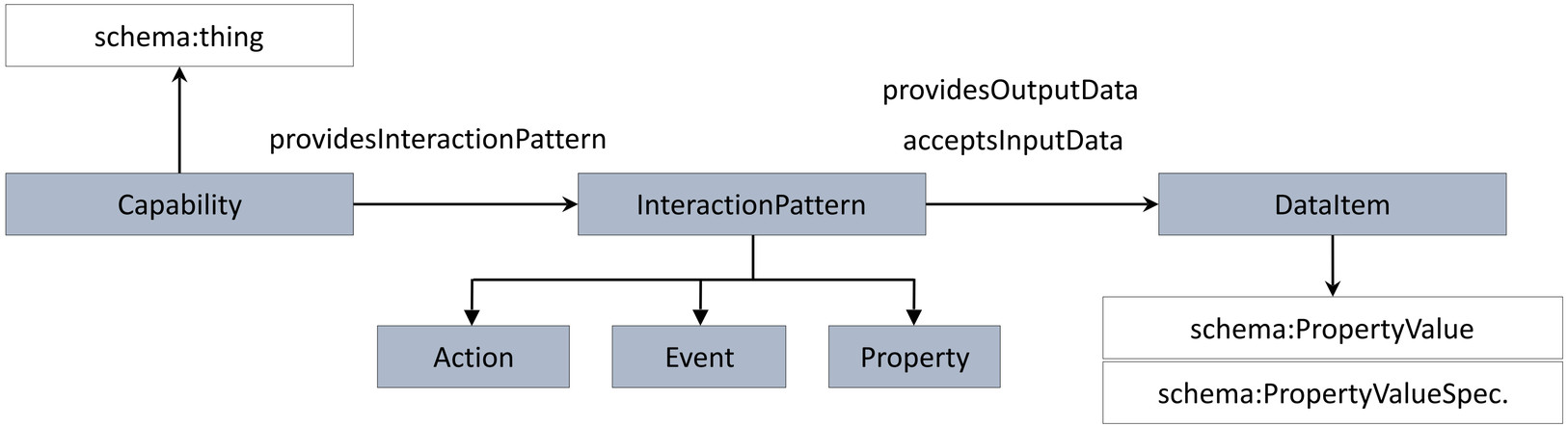
iot.schema.org Update

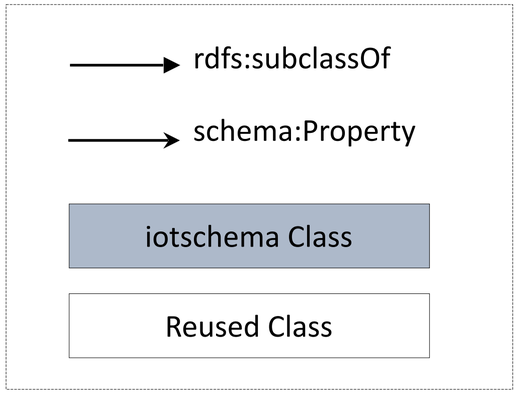
- About 1 year of experience since the WISHI workshop in Prague, July 2017
- Validating the basic categories and annotation style in WoT Thing Description annotation
- Used in WoT Plugfests and WISHI hackathons
- 20-30 initial experimental definitions
- Feature of Interest pattern added for physical world context

Example Semantic Annotation

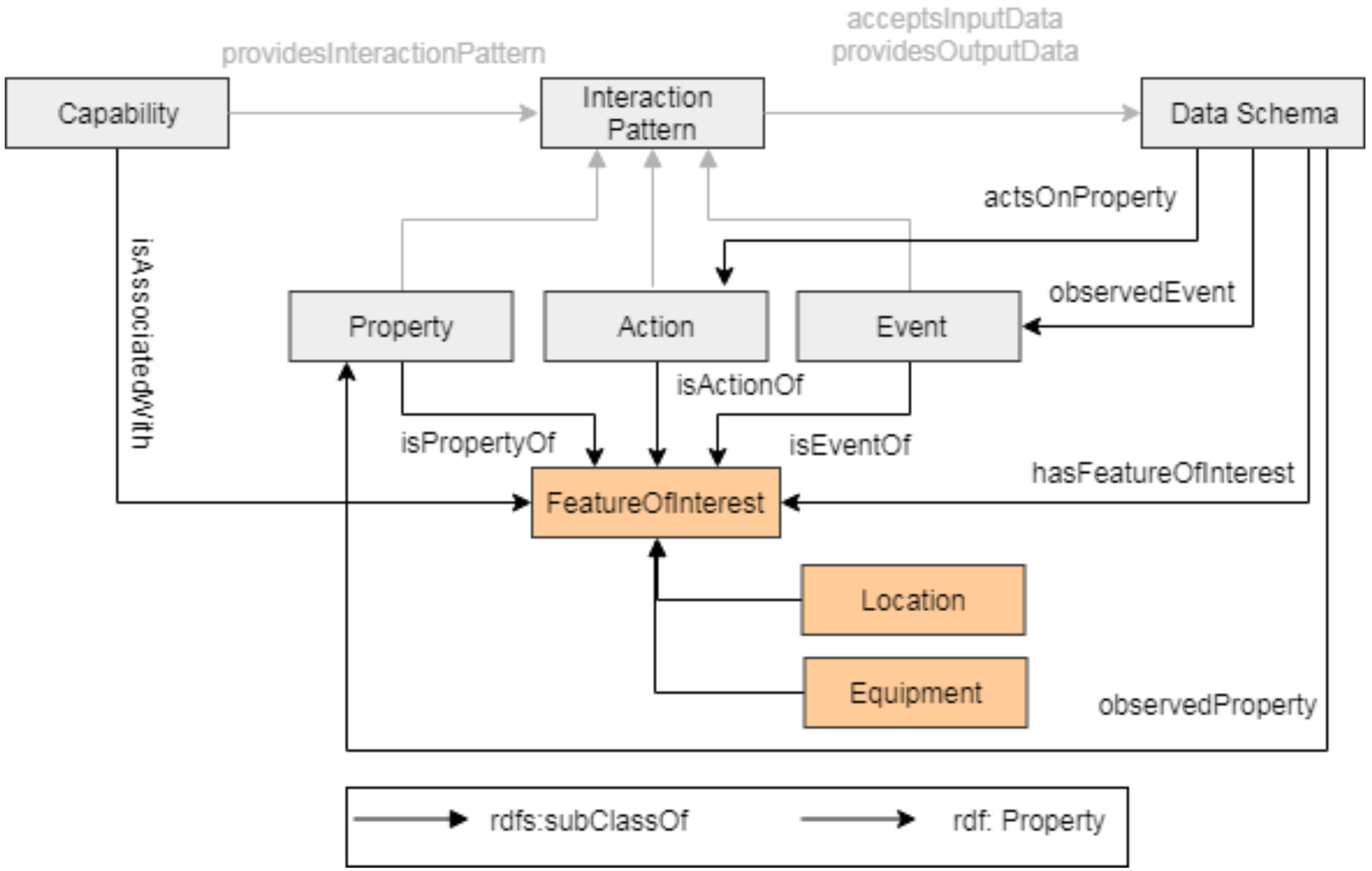
```
{
  "@context": [
    "http://w3c.github.io/wot/w3c-wot-td-context.jsonld",
    "http://w3c.github.io/wot/w3c-wot-common-context.jsonld",
  {"iot": "http://iotschema.org/"}
  ],
  "base": "coap://example.net:5683/",
  "@type": [ "Thing", "iot:TemperatureCapability" ],
  "name": "Temperature Sensor",
  "interaction": [
    {
      "name": "Temperature",
      "@type": ["Property", "iot:Temperature"],
      "outputData":  {
        "type": "object",
        "field": [
          {
            "name": "temperature",
            "@type": ["iot:TemperatureData"],
            "type": "number",
            "minimum": -50,
            "maximum": 100,
            "unit": "Celsius"
          }
        ]
      }
```

iot.schema.org Semantic Categories



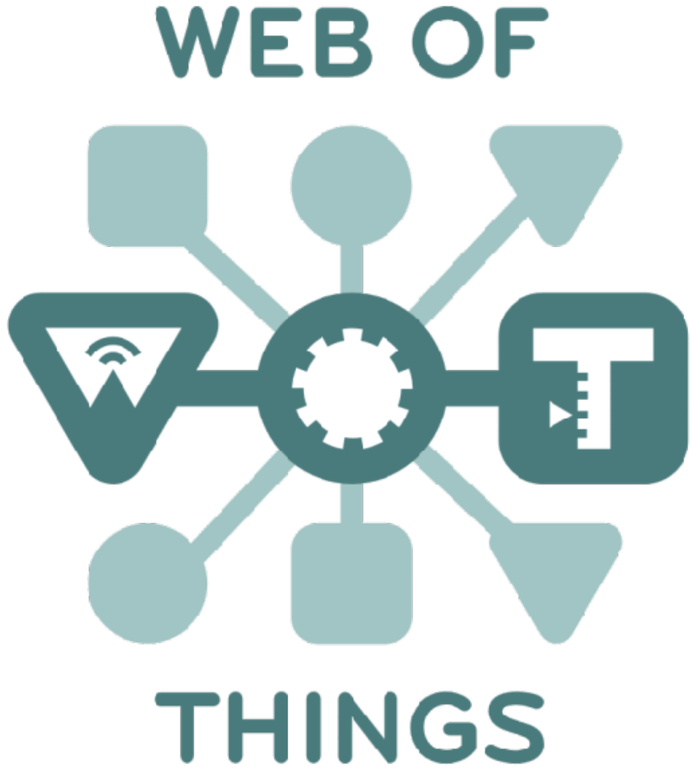


Feature Of Interest Pattern



iot.schema.org Roadmap

- Developing a process to accept contributions
  - W3C CG for vocabulary incubation
  - Github PR to "incoming" folder
  - CI Validation
  - Review and acceptance
  - Publication on iot.schema.org
- Create introductory materials
- Create tools to help build and use definitions
- Build out Feature of Interest
- Enable multiple vertical application domains
- Monthly Community Teleconference



W3C WoT Update

IETF 102, T2TRG, Montreal, Canada, July 2018

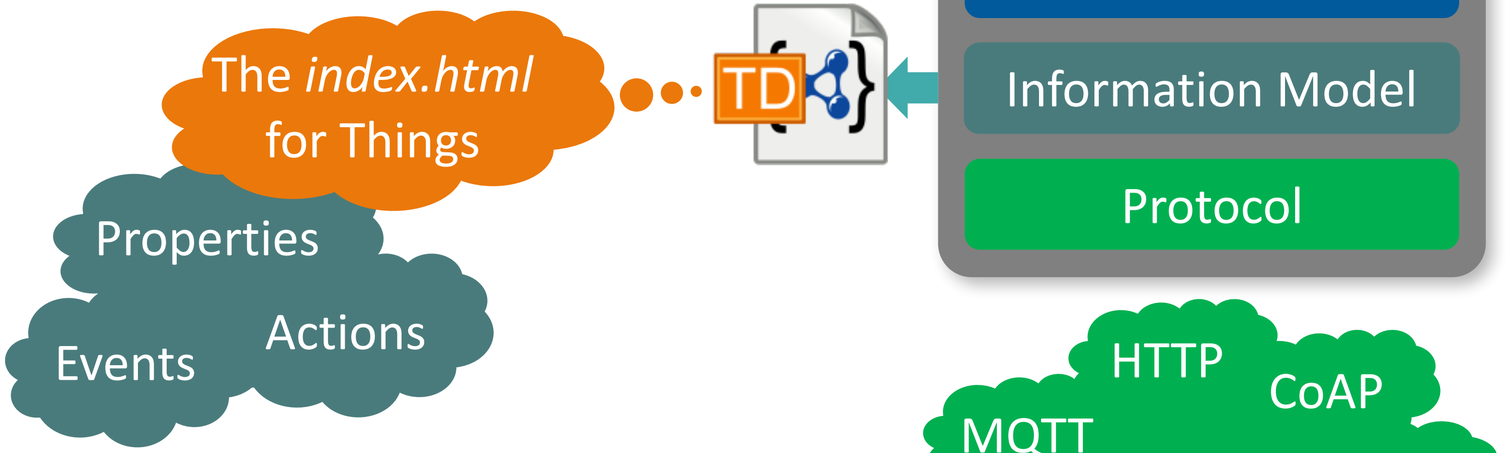
W3C Web of Things – Summary

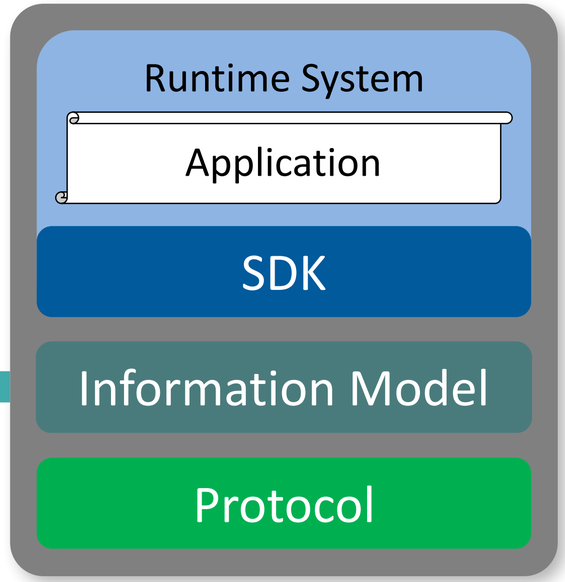
- Counter fragmentation in the IoT
  - Web of Things (WoT) vs Internet of Things (IoT) is similar to World Wide Web vs Internet
  - Take patterns from the World Wide Web and adapt and apply them to the IoT
    - JSON, Schema, and Linked Data
    - URIs and Media Types
    - JavaScript runtime
- By Describing and Complementing
  - Not competing with existing IoT standards, as not prescribing a full-stack solution
  - W3C WoT offers building blocks to pick that enable semantic interoperability
    - WoT Thing Description (TD)
    - WoT Binding Templates
    - WoT Scripting API

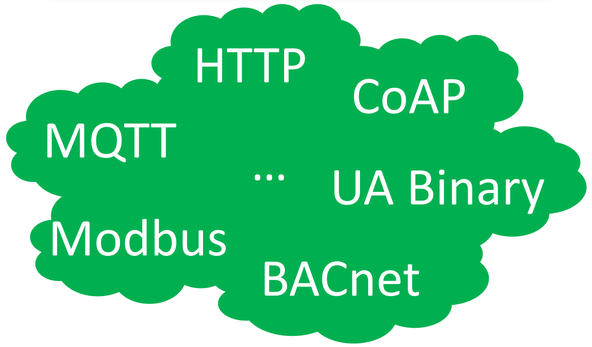
W3C Web of Things – Building Block Approach

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 and interpret their data/services.







JavaScript

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.

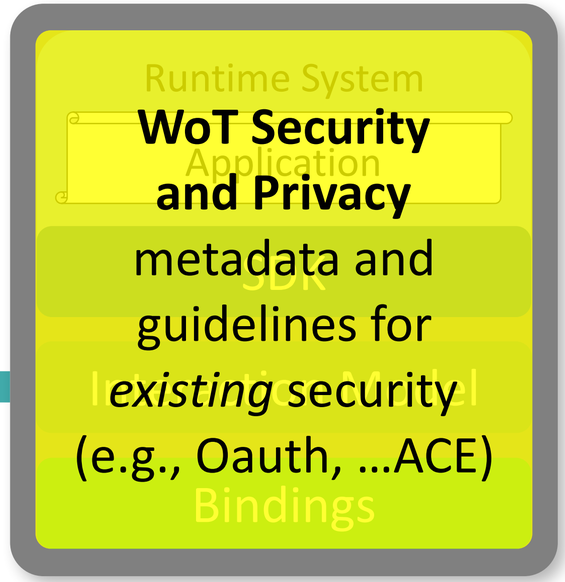
WoT Binding Templates

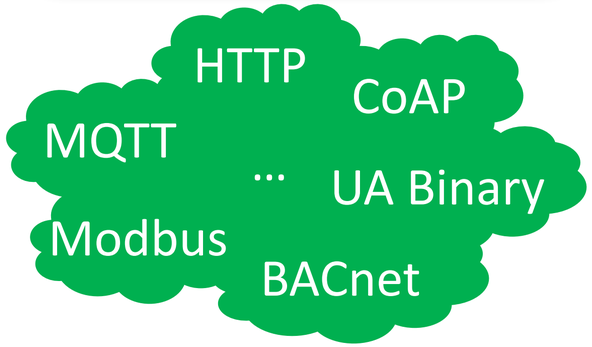
Mappings of the **formal Interaction Model** to concrete protocol operations (e.g., CoAP) and platform features (e.g., OCF). Existing templates are used to easily produce TDs for the Things of the corresponding platform.

W3C WoT Approach – Batteries Included

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 and interpret their data/services.





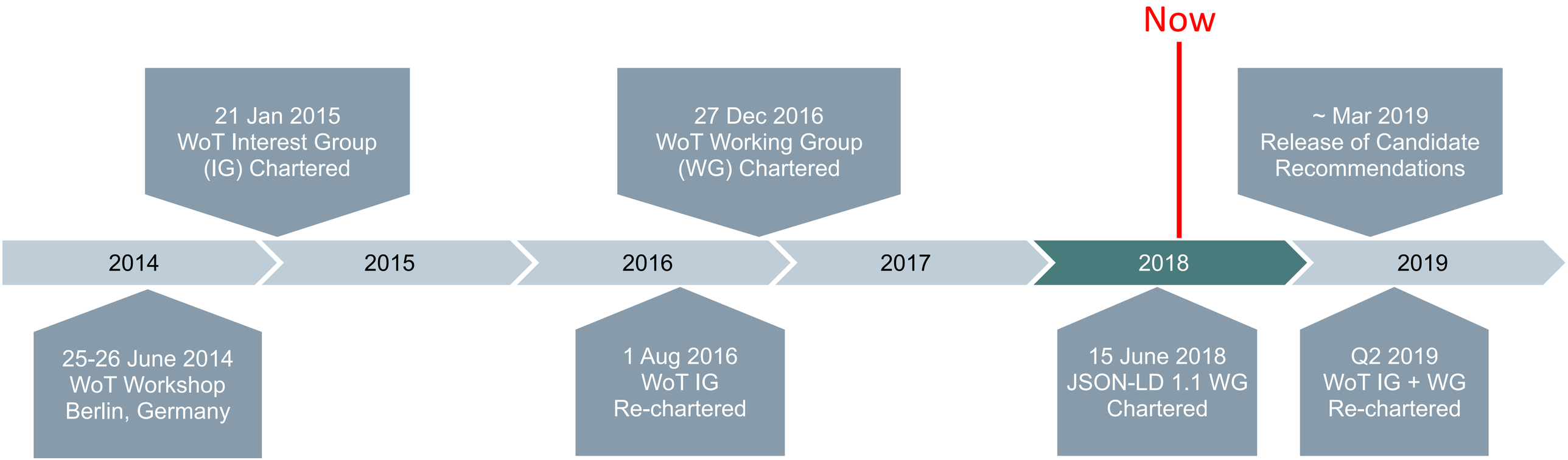
WoT Scripting API

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

Mappings of the **formal Interaction Model** to concrete protocol operations (e.g., CoAP) and platform features (e.g., OCF). Existing templates are used to easily produce TDs for the Things of the corresponding platform.

W3C Web of Things – Timeline



Changed to “Simplified TD” in March 2018

- **JSON-LD 1.1** processing
  - Objects instead of arrays (“idiomatic JSON”)
  - Default values (e.g., `"writable": false`)
  - Framing to serialize and preprocess
- Semantic annotations optional
  - TDs can be treated as simple JSON format
  - New Media Type `application/td+json`
  - Context and terms known via media type
  - No JSON-LD keywords or processing required
  - No LD convention of terms being singular
  - `properties`, `actions`, `events` on top level
- JSON Schema compatibility
  - Data schema syntax now also identical
  - Payloads can be validated directly with JSON Schema implementations
- New terms
  - `id` (as mandatory)
  - `description`
  - `support`
  - … collecting more

Changes in “Simplified TD”

OLD

```
{
  "@context": "https://./w3c-wot-td-context.jsonld",
  "name": "Lamp",
  "base": "coaps://servient.example.com/things/lamp/",
  "interaction": [{
    "@type": "Property",
    "name": "on",
    "schema": { "type": "boolean" },
    "writable": false,
    "observable": false,
    "form": [{
      "href": "properties/on",
      "mediaType": "application/cbor"
    }]
  }, {
    "@type": "Property",
    "name": "brightness",
    "writable": true,
    "observable": false,
    "schema": {
      "type": " integer",
      "minimum": 0,
      "maximum": 100
    },
    "form": [{
      "href": "properties/status",
      "mediaType": "application/cbor"
    }]
  }, {
    "@type": "Action",
    "name": "fade",
    "inputSchema": {
      "type": "object",
      "fields": [{
        "name": "from",
        "schema": {
          "type": "integer",
          "minimum": 0,
          "maximum": 100
        }
      }, {
        "name": "to",
        "schema": {
          "type": "integer",
          "minimum": 0,
          "maximum": 100
        }
      }, {
        "name": "duration",
        "schema": { "type": "number" }
      }]
    },
    "forms": [{ "href": "/things/lamp/actions/fade" }]
```

NEW

```
{
  "id": "urn:dev:ops:32473-smartlight-4711",
  "name": "Lamp",
  "description": "Corner torchiere“,
  "base": "coaps://servient.example.com/things/lamp/",
  "properties": {
    "on": {
      "type": "boolean",
      "forms": [{
        "href": "properties/on",
        "mediaType": "application/cbor"
      }]
    },
    "brightness": {
      "type": "integer",
      "minimum": 0,
      "maximum": 100
      "writable": true,
      "forms": [{
        "href": "properties/status",
        "mediaType": "application/cbor"
      }]
    }
  },


  "actions": {
    "fade": {
      "input": {
        "type": "object",
        "properties": {
          "from": {
            "type": "integer",
            "minimum": 0,
            "maximum": 100
          },
          "to": {
            "type": "integer",
            "minimum": 0,
            "maximum": 100
          },
          "duration": { "type": "number" }
        }
      },
      "forms": [{
        "href": "/things/lamp/actions/fade",
        /* encType would be for the request body
           opposed to mediaType, which is for target
           FIXME: can have both meanings based on context (links/forms)? */
        "encType": "application/json",
        "mediaType": "application/json"
      }]
```

Changes in “Simplified TD”

OLD

```
  }, {
    "@type": "Action",
    "name": "fade",
    "inputSchema": {
      "type": "object",
      "fields": [{
        "name": "to",
        "schema": {
          "type": "integer",
          "minimum": 0,
          "maximum": 100
        }
      }, {
        "name": "duration",
        "schema": { "type": "number" }
      }]
    },
    "form": [{
      "href": "actions/fade",
      "mediaType": "application/cbor"
    }]
  },


  {
    "@type": "Event",
    "name": "overheated",
    "schema": {
      "type": "object",
      "fields": [{
        "name": "temperature",
        "schema": { "type": "number" }
      }]
    },
    "forms": [{ "href": "/things/lamp/events/overheated" }]
  }],
  "links": [{
    "href": "https://servient.example.com/things/motion-detector",
    "rel": "controlledBy",
    "mediaType": "application/td"
  }]
}
```

NEW

```
  "actions": {
    "fade": {
      "input": {
        "type": "object",
        "properties": {
          "to": {
            "type": "integer",
            "minimum": 0,
            "maximum": 100
          },
          "duration": { "type": "number" }
        }
      },
      "forms": [{
        "href": "actions/fade",
        "mediaType": "application/cbor",
        "inputMediaType": "application/cbor"
      }]
    }
  },


  "events": {
    "overheated": {
      "type": "object",
      "properties": {
        "temperature": { "type": "number" }
      },
      "forms": [{
        "href": "/things/lamp/events/overheated",
        /* needed, alternative: register URI schemes "http+sse", "http+lp", ... */
        "http:subProtocol": "http:EventSource",
        "mediaType": "application/json"
      }]
    }
  },
  "links": [{
    "href": "https://servient.example.com/things/motion-detector",
    "rel": "controlledBy",
    "mediaType": "application/td"
  }]
}
```

Changes in “Simplified TD”

OLD

```
}, {
    "@type": "Event",
    "name": "overheated",
    "schema": {
      "type": "object",
      "fields": [{
        "name": "temperature",
        "schema": { "type": "number" }
      }]
    },
    "form": [{
      "href": "https://.../events/overheated",
      "mediaType": "application/json"
    }]
  }],
  "link": [{
    "href": "https://servient.example.com/things/pir",
    "rel": "controlledBy",
    "mediaType": "application/ld+json"
  }]
}
```

NEW

```
  "events": {
    "overheated": {
      "type": "object",
      "properties": {
        "temperature": { "type": "number" }
      },
      "forms": [{
        "href": "https://.../events/overheated",
        "subProtocol": "LongPoll",
        "mediaType": "application/json"
      }]
    }
  },


  "links": [{
    "href": "https://servient.example.com/things/pir",
    "rel": "controlledBy",
    "mediaType": "application/td+json"
  }]
}
```

Changes in Scripting API

OLD

```
let thing = WoT.produce({
    name: "counter"
    // no support for
    // more metadata
});

console.log("Created thing " + thing.name);

thing.addProperty(
  {
    name : "count",
    schema : '{ "type": "number"}',
    // no support for
    // custom metadata
    observable : true,
    writeable : true,
    value : 0
  });

thing.addAction({ name: "increment" });
thing.setActionHandler(
  "increment",
  () => {
    return thing.readProperty("count").then(res=>{
      thing.writeProperty("count", ++res);
    });
  });
```

NEW

```
let thing = WoT.produce({
    name: "counter",
    description: "counter example Thing",
    "@context": { "iot": "http://iotschema.org/" }
  });

  console.log("Created thing " + thing.name);

  thing.addProperty(
    "count",
    {
      type: "integer",
      description: "current counter value",
      "iot:Custom": "example annotation",
      observable: true,
      writeable: true
    },
    0);

  thing.addAction("increment");
  thing.setActionHandler(
    "increment",
    () => {
      return thing.properties["count"].get().then(res=>{
        thing.properties["count"].set(++res);
      });
    });
```

Changes in Scripting API

OLD

```
WoT.fetch("http://localhost:8080/counter")
  .then(td => {

    let thing = WoT.consume(td);

    // introspection had to parse td
    // in application code

    thing.readProperty("count")
      .then(res => {
        console.info("count value is", res);
      })
      .catch(err => { console.error(err); });

    thing.invokeAction("increment");

  })
  .catch(err => { console.error(err); });
```

NEW

```
WoT.fetch("http://localhost:8080/counter")
  .then(td => {

    let thing = WoT.consume(td);

    // introspection support (type, desc., iot:Custom, …)
    console.dir(thing.properties.count);

    thing.properties.count.get()
      .then(res => {
        console.info("count value is", res);
      })
      .catch(err => { console.error(err); });

    thing.actions.increment.invoke();

  })
  .catch(err => { console.error(err); });
```

W3C Web of Things – Todos until Release

- WoT Thing Description
  - Extend model to efficiently support read-/write-multiple interactions
  - Revisit Events to allow for input on subscribe (e.g., filters)
  - Finalize model for security vocabulary
  - Align `links` field with draft-ietf-core-links-json
    - Found issue with `type` attribute lacking parameter support
  - Collect more core vocabulary terms (e.g., `version`, `created`, `lastModified`)
  - IANA Considerations
    - application/td+json and CoAP Content-Format number

W3C Web of Things – Todos until Release

- WoT Scripting API
  - Model read-/write-multiple interactions in the API
  - Finalize discovery
  - Define API errors
- WoT Binding Templates
  - Create extension point for hypermedia-driven Actions and Events
    - application/wot+json and CoAP Content-Format number
- WoT Security and Privacy
  - Refine initial but extensible security vocabulary (based on TD model)
  - Start a “living” Working Group Note on “WoT Security Best Practices”

Contact

Dr. Matthias Kovatsch

Senior Research Scientist

Siemens AG

CT RDA IOT EWT-DE

matthias.kovatsch@siemens.com

Next Steps in Security

- Oscar Garcia-Morchon: Automated IoT Security
- Mohit Sethi: Enabling Network Access for IoT devices from the Cloud
- René Struik: Next Steps in Security
- Dirk Kutscher: Decentralized Trust for IoT and In-Network-Computing
- Carsten Bormann: IoT Security Semantics and Semantics Security

Automated IoT Security

T2TRG - IETF102 - Montreal
19/07/2018

Oscar Garcia-Morchon (Philips)

Goal of the Draft

https://datatracker.ietf.org/doc/draft-garciamorchon-t2trg-automated-iot-security/

Solving the mismatch between

- The security capabilities and settings with which IoT devices are designed / manufactured / deployed
- The actual security requirements of the IoT devices in different environments over time

Work derived from the “State-of-the-Art and Challenges for the Internet of Things Security” document:
https://datatracker.ietf.org/doc/draft-irtf-t2trg-iot-seccons/

Problems to solve

Problem 1: Different environments

- Deploying in a home is not the same as in an office or in the Department of Defense

Problem 2: Evolving threats

- Algorithms become insecure
- Bugs in software are found
- Users change their preferences

Problem 3: Pre-configuration is not always right

- a product owner doesn’t know they should disable a protocol;
- a developer doesn’t remove all of the off ending code (just some uses of it);
- the documentation doesn’t mention the protocol, even though the device implements it;

Overview

- Part 1: Examples of Security Threats and Mitigation strategies for IoT
  - Firmware Replacement
  - Extraction of private information
  - Data leakage - cryptographic keys

- Part 2: Security framework to include existing risk and vulnerability assessment processes
  - Business Impact Analysis
  - Risk Assessment
  - Privacy Impact Assessment
  - Vulnerability Assessment
  - Incident Reporting

  during the lifecycle of a smart object

Overview

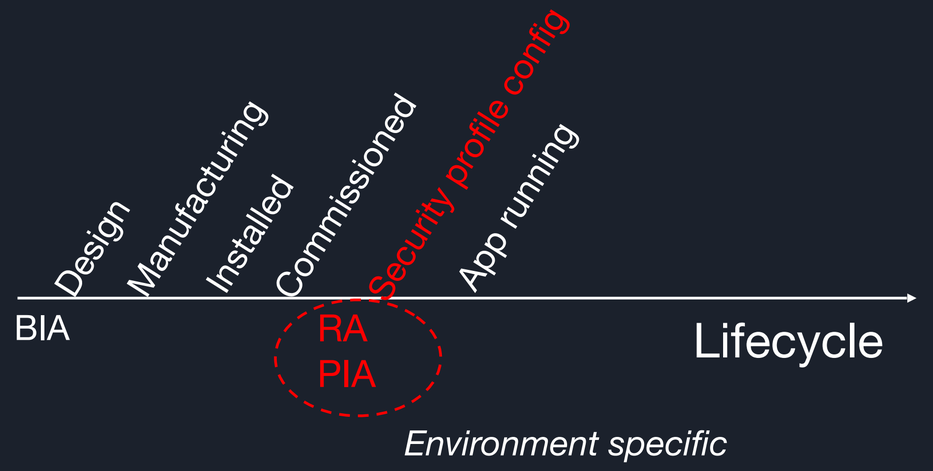
- Part 3: Security Profiles and application to IoT devices in a specific environment including
  - a short descriptive name
  - an exemplary application that might use the security profile
  - the main security threats applicable to the profile
  - the security mitigations required by the profile
  - specific configuration parameters for the protocols and actors involved in the application

PASC - Protocol for Automatic Security Configuration

Enabling automatic security configuration of Things by shifting methodologies for risk management from the tailored product design and implementation phases to the onboarding phase

Current practice

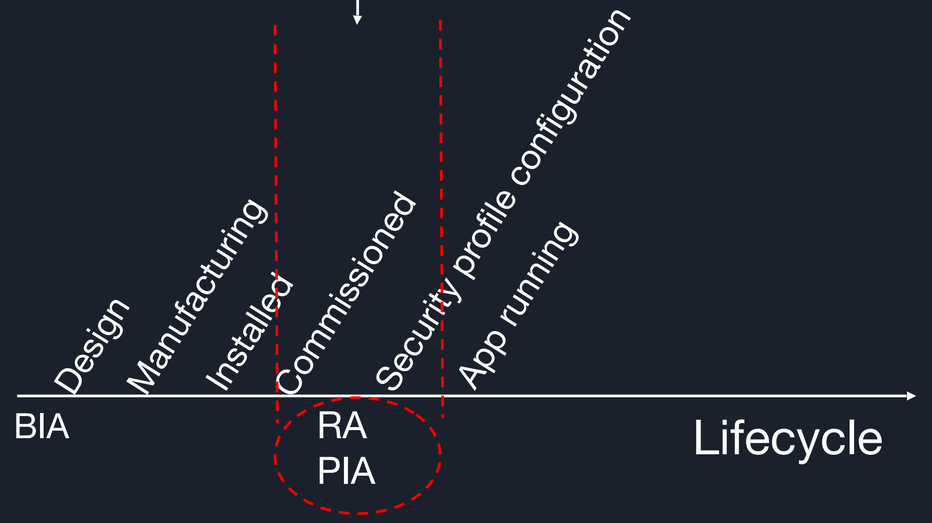
PASC: 1st protocol in our draft



PASC - Protocol for Automatic Security Configuration

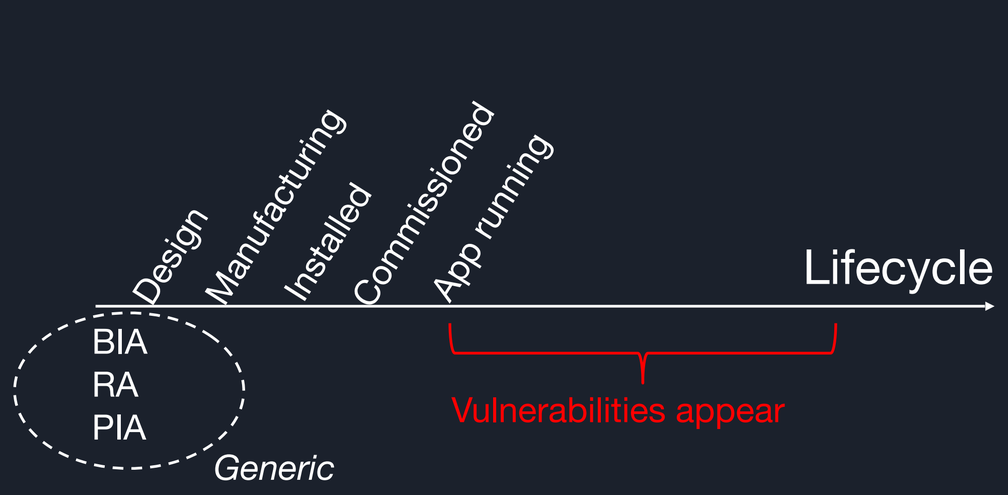
High-level idea of PASC

- Thing to publish its usage profile to a Gateway
- Gateway gathers additional information about the Thing, the usage and expected interactions of the smart object with other devices in the deployment environment (e. g. via MUD, portscan)
- Gateway performs an automated risk assessment
  - Determines potential threats on the device and on deployment environment
  - Determines security profile containing mitigations
- Deploy updated security profiles
  - to the Thing itself
  - to other devices already present in the deployment environment (other smart objects, Firewalls)

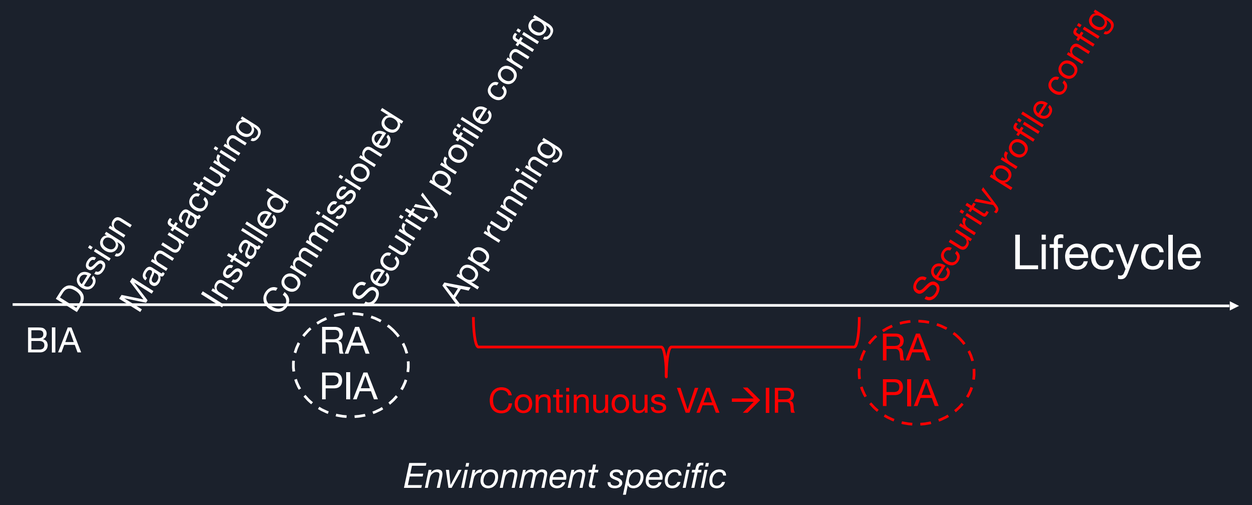


PAVA - Protocol for Automatic Vulnerability Assessment

Current practice



PAVA: Second protocol in our draft



PAVA - Protocol for Automatic Vulnerability Assessment

High-level idea of PAVA

- Thing to send standardized reports of potential vulnerabilities to a Gateway via Syslog
- Gateway to analyse the reports and decide regarding the existence of a vulnerability, its origin and its impact
- Gateway to run additional and continuous analysis of each Thing based on Security Profile

Enabling updates of security profiles in real time and automatic incident reporting towards

- the user
- the manufacturer
- the deployment environment provider

Benefits

- Benefits for manufacturers
  - no need to decide which security mitigations are required for each product
  - simply describe the expected usage of the Thing

- Benefits for system operators
  - minimize operational cost while ensuring that the system remains secure at any moment
  - enabling automation for security configuration in deployment environments with potentially millions of smart Things

- Benefits for end users
  - security configuration is done in an automatic way
  - users “don’t need to do anything”

IETF T2TRG: Enabling Network Access for IoT devices from the Cloud

IETF 102

19 July, 2018

Montreal

Authorizing network access for IoT devices

- New off-the-shelf devices need Internet access
  - for vendor and third-party services in the cloud
  - for software update

Authorizing network access for IoT devices

Two problems:

- Discovery and configuration: which network?
  - For example, need to find the right SSID and cloud server
- Security bootstrapping: identifiers and credentials?
  - For connecting to the network
  - For connecting to the cloud

Authorizing network access for IoT devices

Challenges:

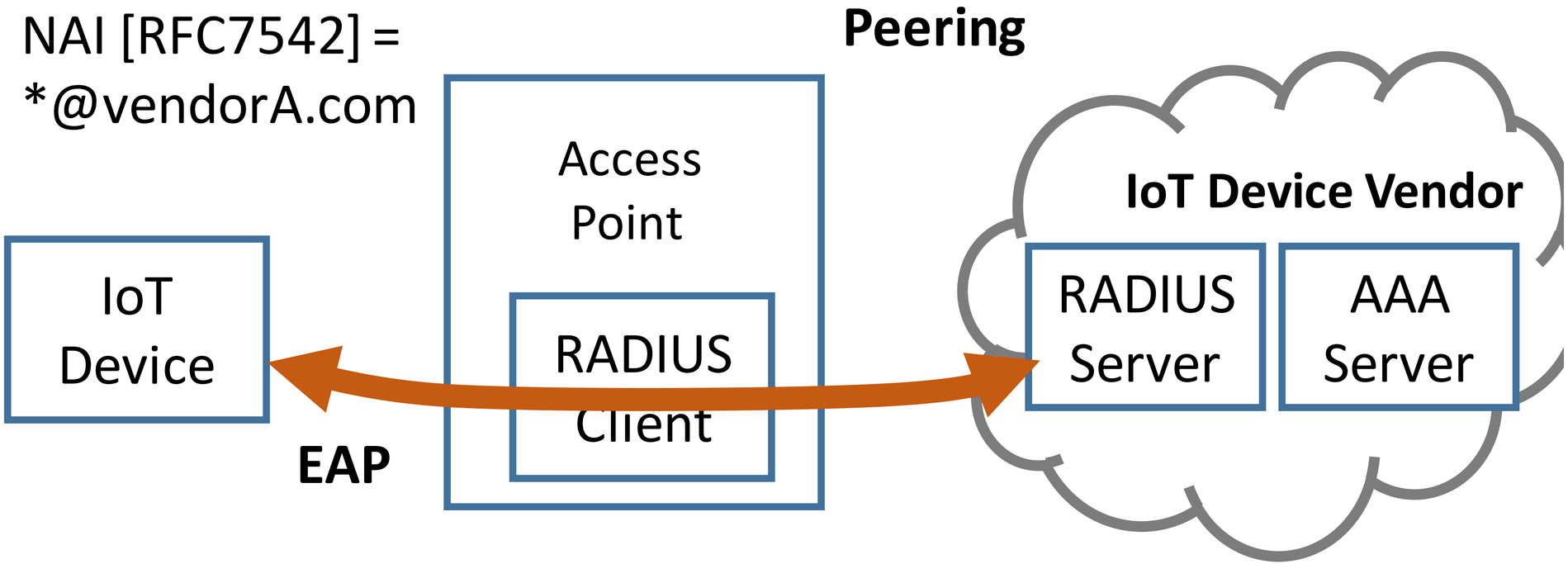
- Limited user interface
- Scalability
- At home, small office, enterprise or industrial environment
  - Clueless users vs. professional admins and support
  - On the other hand, same devices everywhere
- Wi-Fi (WPA-Personal and WPA Enterprise), Zigbee, BTLE

Authorizing network access for IoT devices

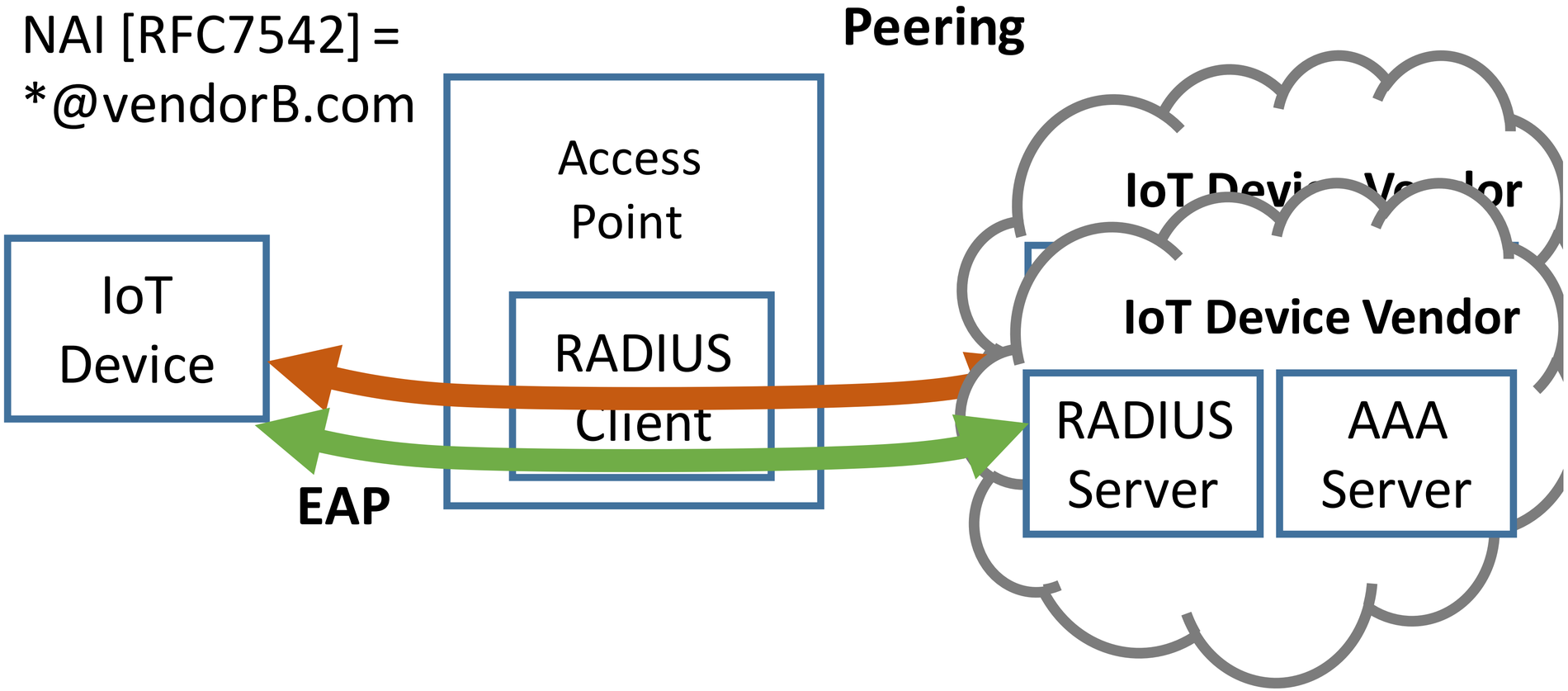
Current Solutions for network access authorization:

- Manual configuration and key distribution
  - Pairing with smart phone over Bluetooth
  - Wifi (Un)Protected Setup (WPS)
- Managed solutions
  - RADIUS / DIAMETER / 802.1x
  - Vendor and enterprise certificates

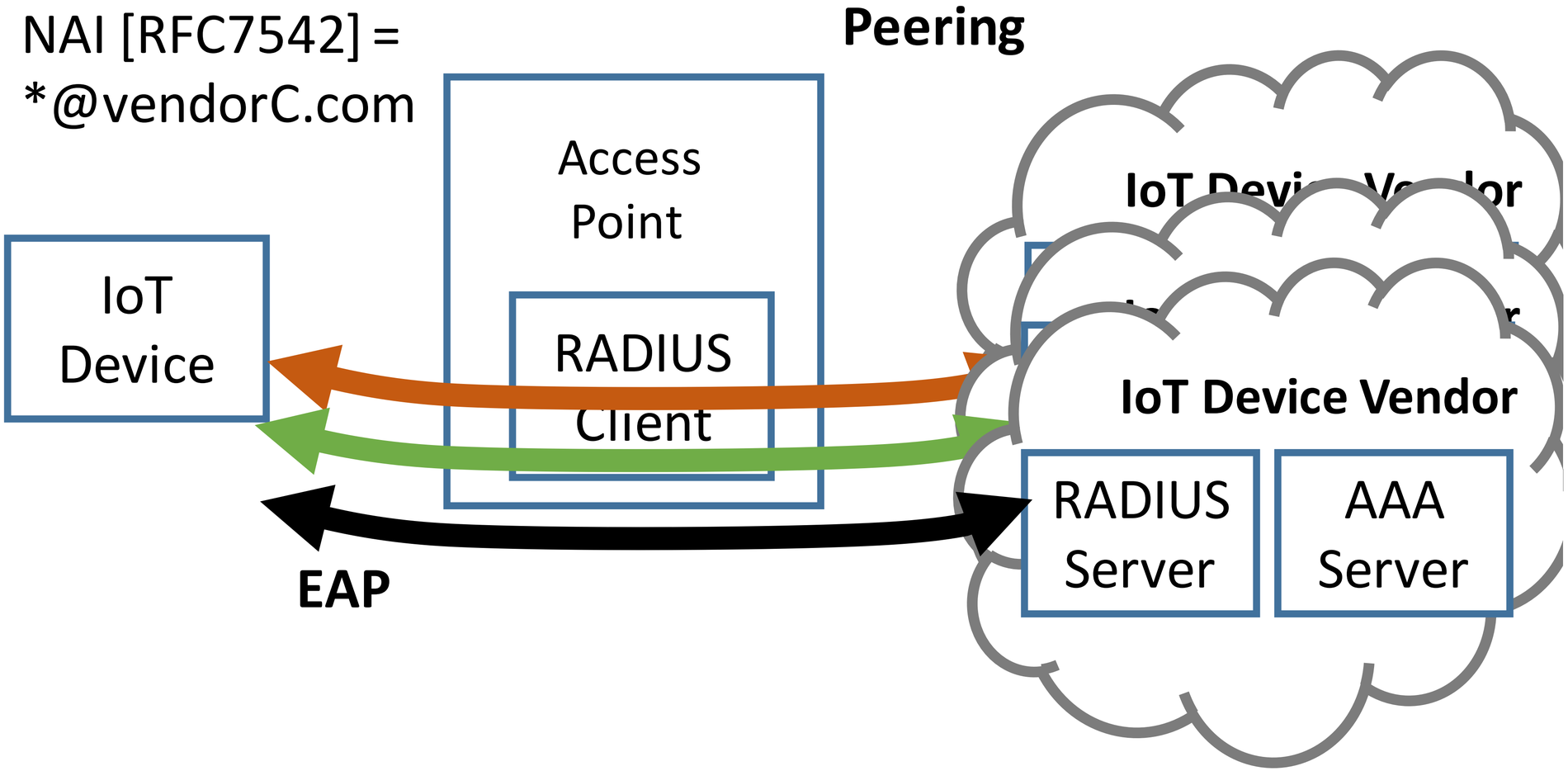
Scenario:
cloud-connected IoT appliance



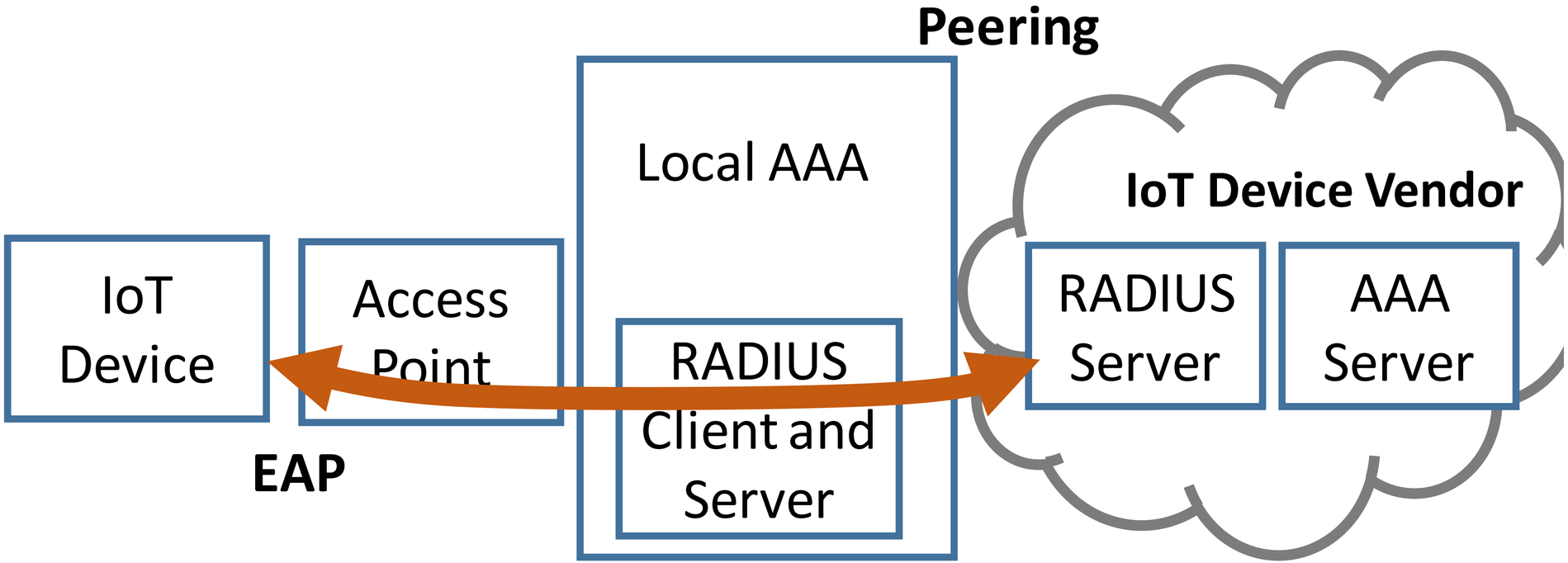
Scenario:
cloud-connected IoT appliance



Scenario:
cloud-connected IoT appliance



Scenario:
cloud-connected IoT appliance



Next Steps in Security?

(Discussion in t2trg)

René Struik

Struik Security Consultancy

E-mail: rstruik.ext@gmail.com

IETF 102 – Montreal, QC, Canada, July 19, 2018

Putting Trust in Devices

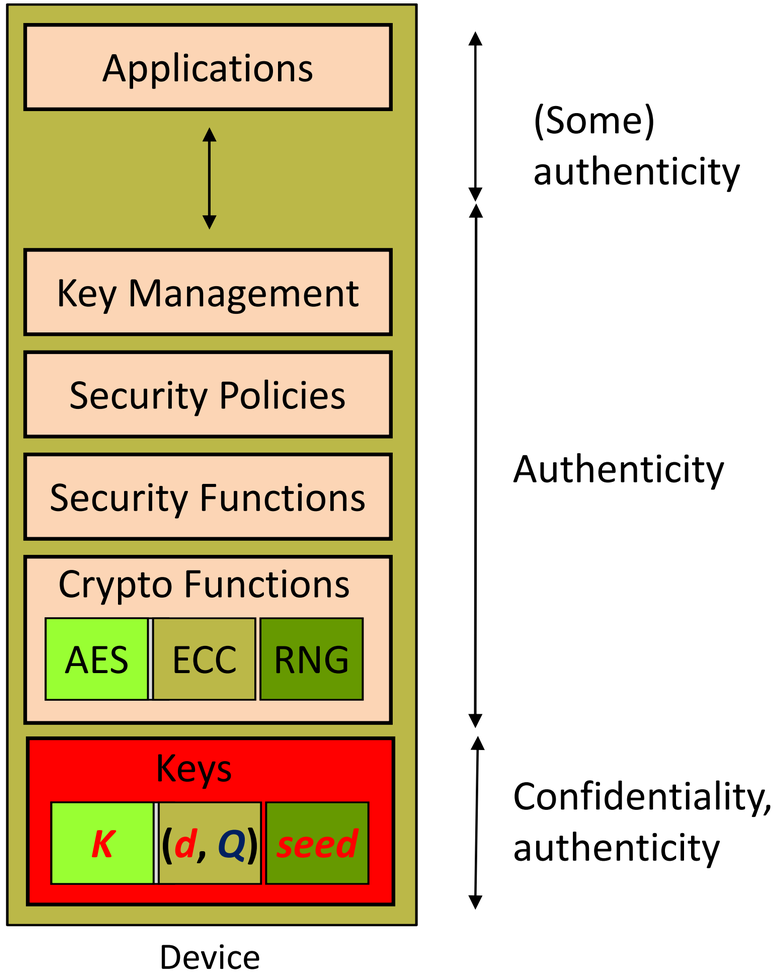
Conventional Approach

- Trusted implementation of crypto, including side channel resistance
- Trusted security policy routines
- Secure and authentic key storage
- Secure RNG (or RNG seed)

Ideal Functionality

- Single function for each task
- Minimizes overall implementation cost

Note:
Seemingly conflicts with "crypto agility"



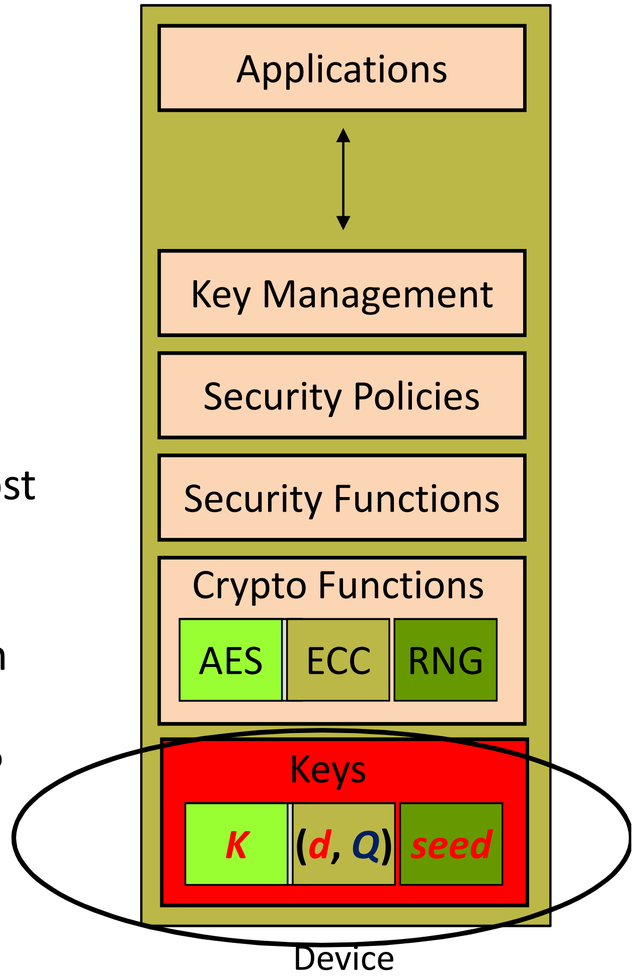
Keying Material

Ideal Functionality

- Single function for each task
- Minimization overall implementation cost

Questions:

- Can one use single public key pair **(*d*, *Q*)** for both key agreement and signing?
  - Single certificate cost, lower key management cost
  - Current perception: “verboten!”
- Can one use single symmetric key ***K***?
  - Different keys provide logical channel separation (however, key hierarchy could lower cost)
- Does one need high-quality random number ***seed***?
  - Long-term keys need high-quality RNG source
  - What about short-term keys, derived keys?
  - What about RNG needs remainder of device?
  - *Distinguish on-device vs. off-device randomness*



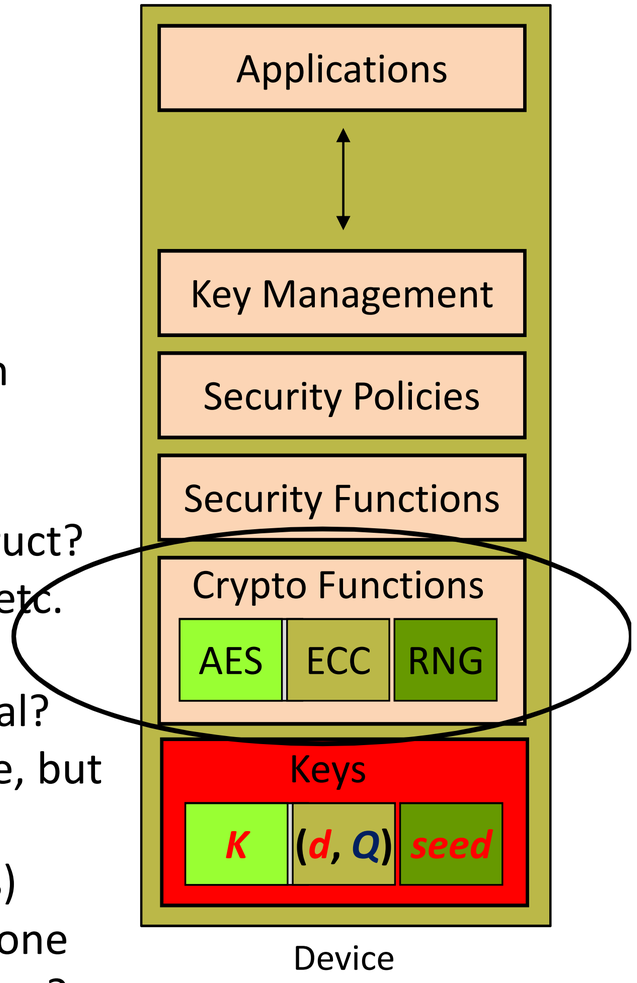
Crypto Functions

Ideal Functionality

- Single function for each task
- Minimization overall implementation cost

Questions:

- Can one reuse existing implementations, even with crypto agility?
  - ECC example: CFRG curves vs. NIST curve
- Can one use single “Swiss Army” symm.-key construct?
  - Block-cipher mode of operation, hash function, etc.
  - Keccak-family to the rescue?
- Are small devices “doomed” should PQ-hype be real?
  - Symmetric-key crypto: not just impact on key size, but also authentication tag length and cryptanalysis (e.g., PQ-distinghuishers in cipher building blocks)
  - Public-key crypto: can passwords help? Or, does one really need (nascent) PQ-schemes with large parms?
  - If classical threats already ignored, why care about PQ…?



Key Management

Ideal Functionality

- Single function for each task
- Minimization overall implementation cost

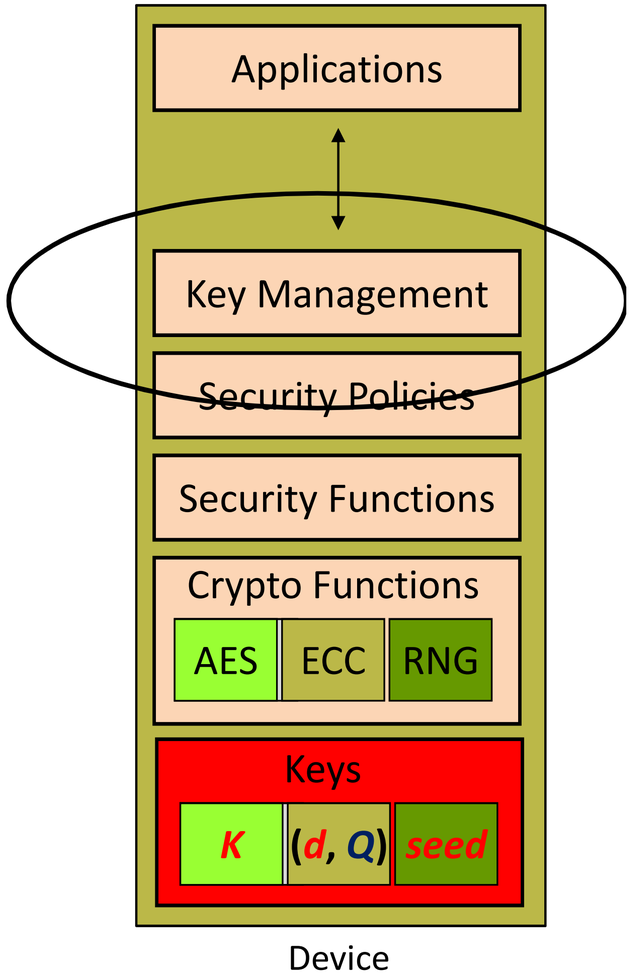
Questions:

- How does one limit impact of key compromise?
  – short-lived certificates at reduced cost?
  – could “ledger” as key repository help?
- What about key provisioning, device configuration and commissioning
- Homo- or heterogeneous devices and networks?

What about privacy and control?

- who owns data?
- what about switching cost?

Note: technology is not neutral here… Should it?



Concluding remarks

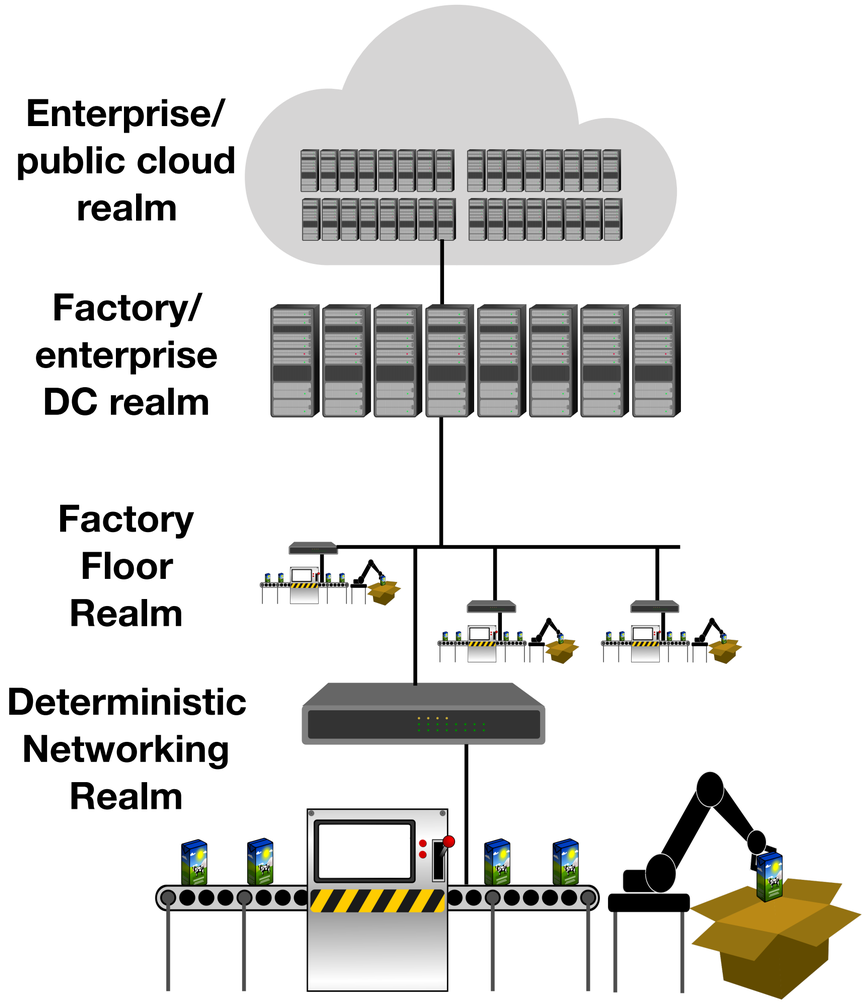
Please reflect on questions as homework assignment …

Decentralized Trust

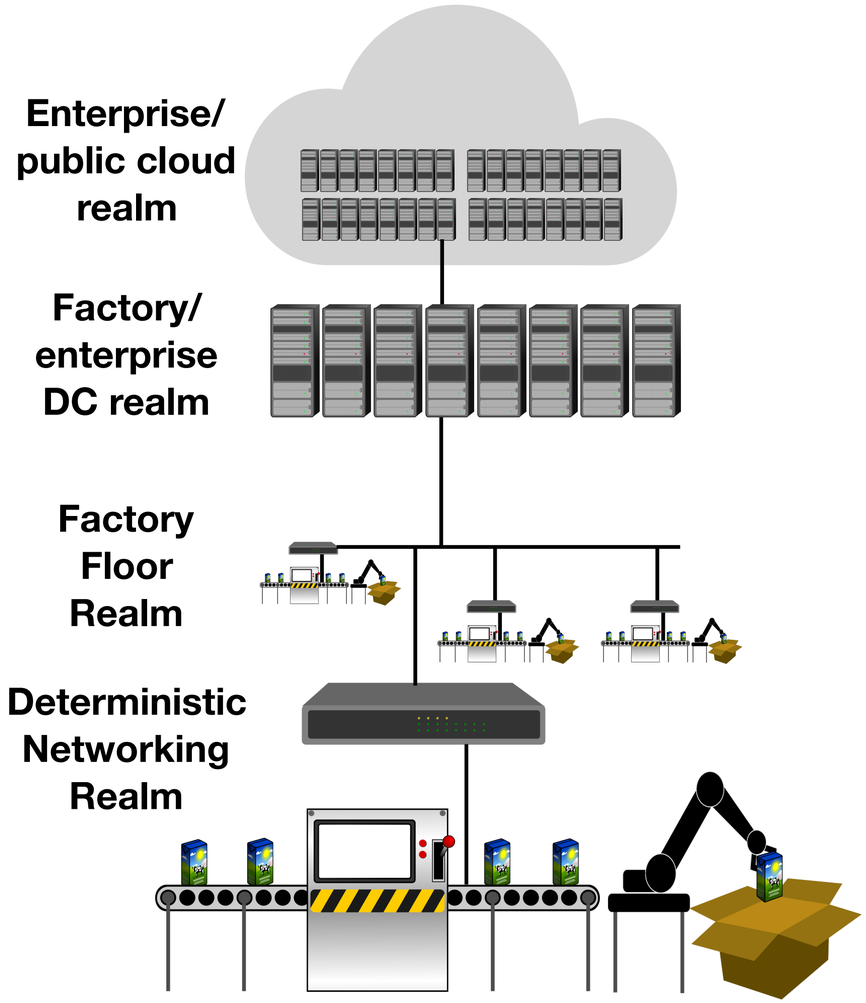
for IoT and In-Network-Computing

Dirk.Kutscher@huawei.com

Industrial IoT



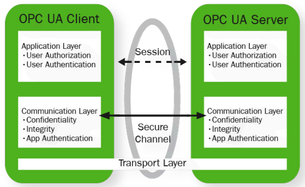
Industrial IoT



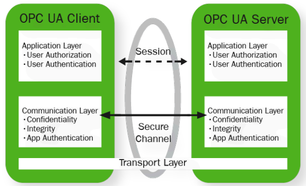
MQTT.ORG

AMQP Advanced Message Queuing Protocol

Data analytics, archival



Cloudified control apps (virtual PLC etc.)



Data exchange & control (OPC UA, DDS)

Ethernet Time Sensitive Networking (TSN)

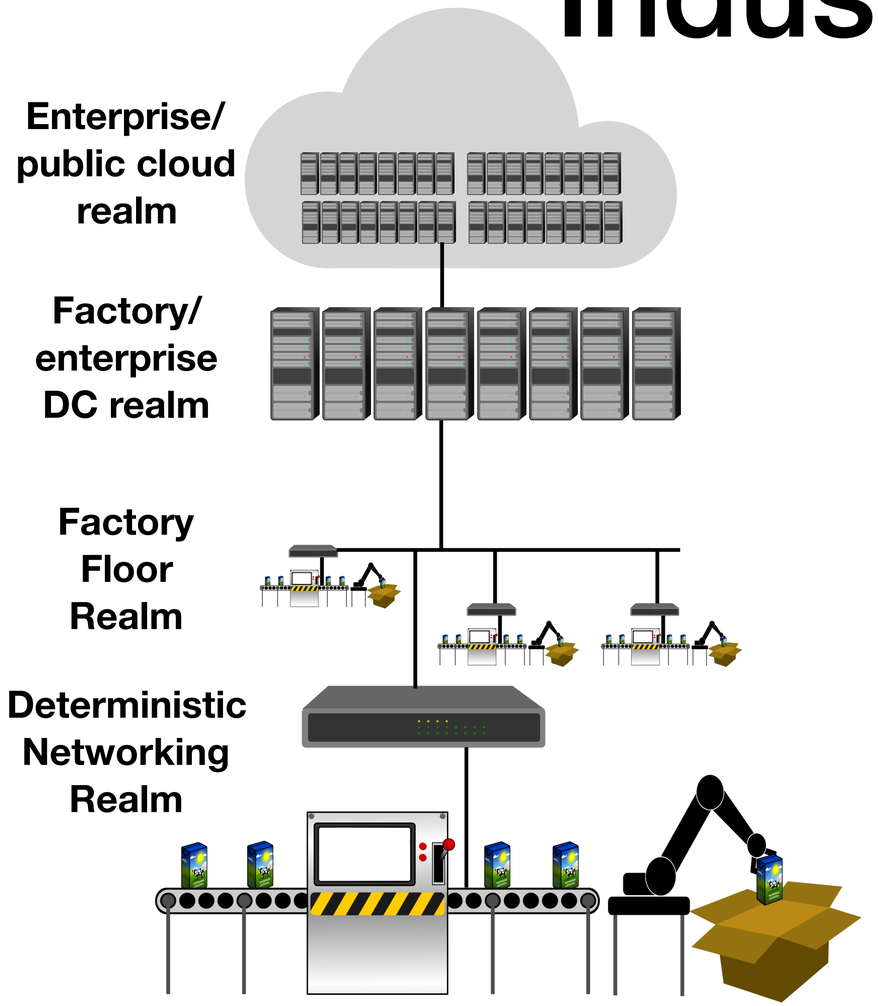
Standard	Description
802.1ASrev	Timing & Synchronization
802.1Qbv	Enhancements for Scheduled Traffic (Timed Gates for Egress Queues)
802.1Qbu	Frame Preemption
802.1Qca	Path Control and Reservation
802.1Qcc	Central Configuration Management
802.1Qci	Per-Stream Time-based Ingress Filtering and Policing
802.1CB	Redundancy, Frame Replication & Elimination

PI PROFIBUS • PROFINET

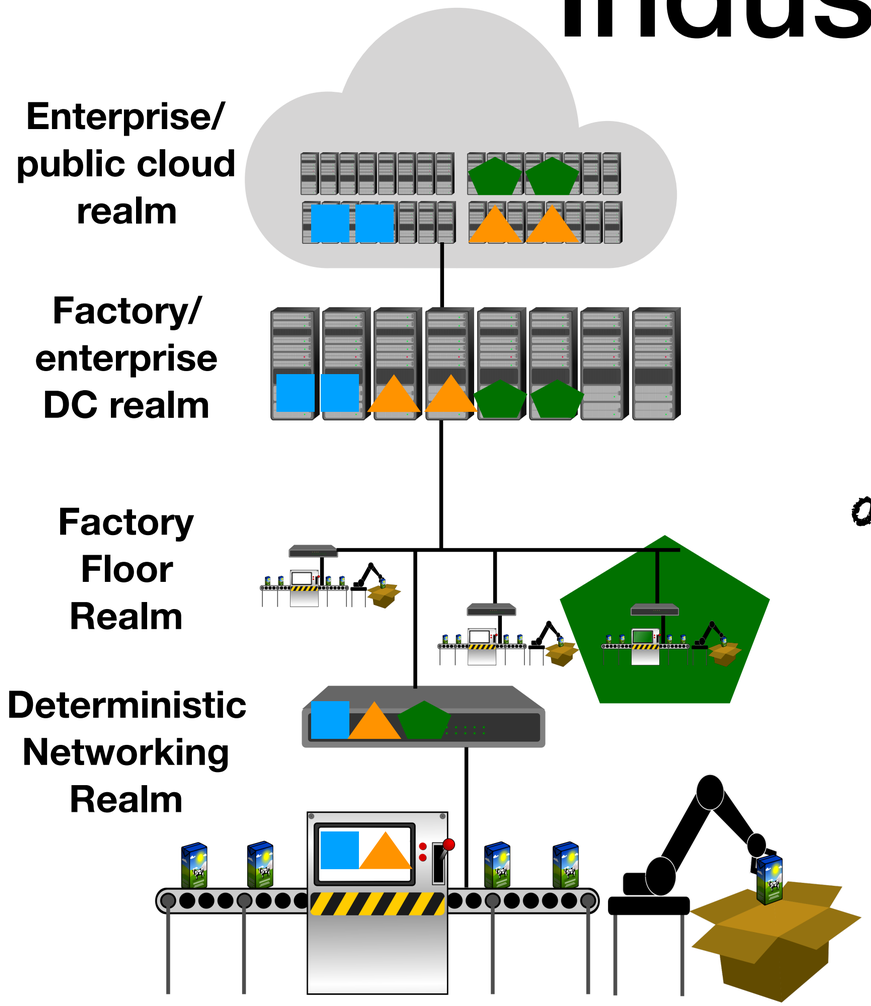
EtherCAT Technology Group

Etc

Vision for Future Industrial IoT

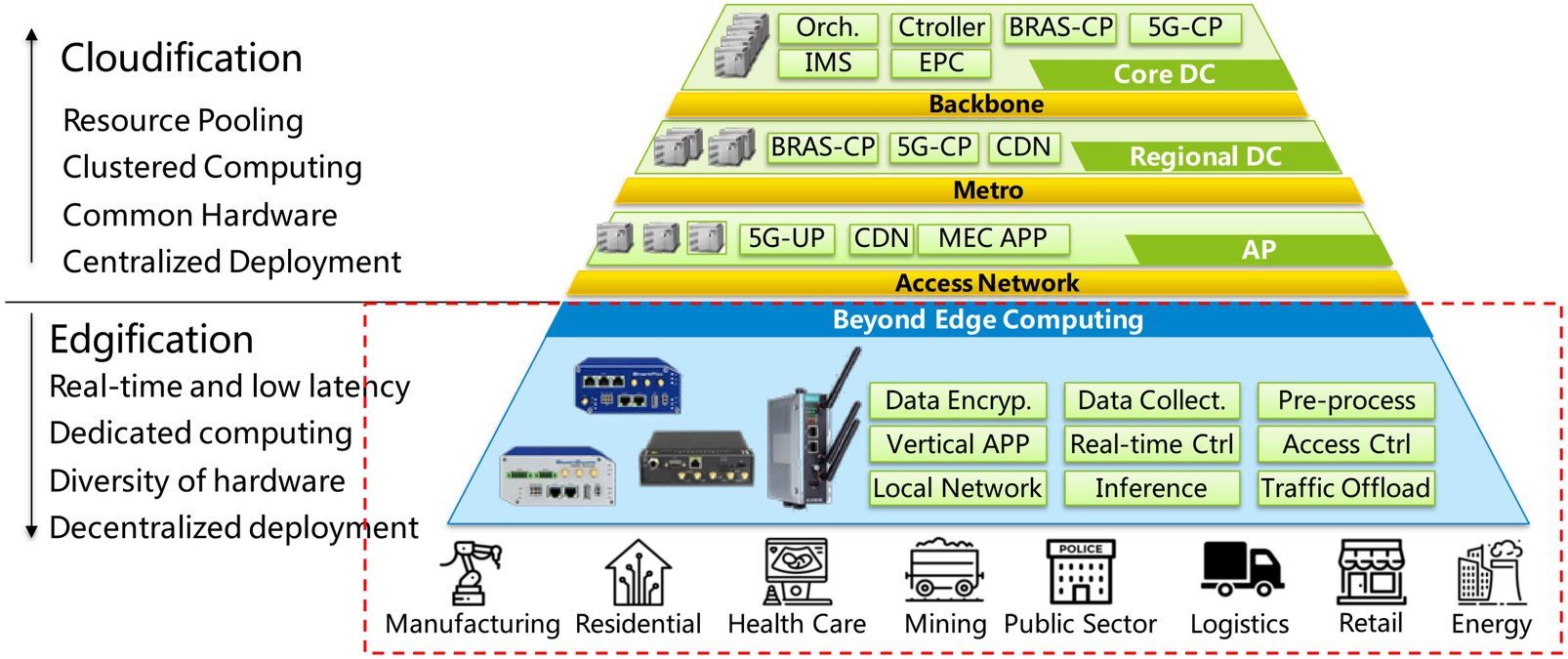


Vision for Future Industrial IoT



Factory/Enterprise network
as a multi-tenant environment

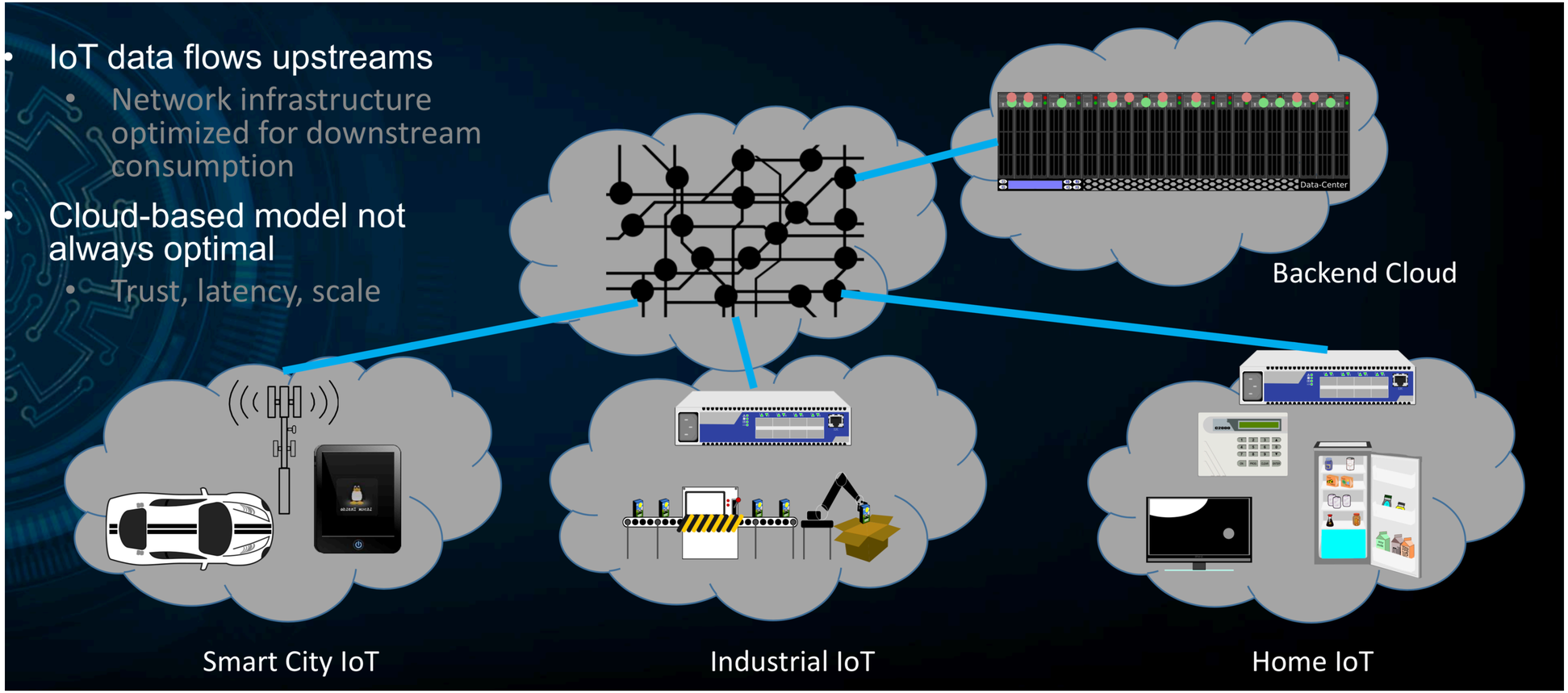
China Mobile: Beyond Edge Computing



BEC takes care of the first hop where the service of a particular industrial vertical connects to the network

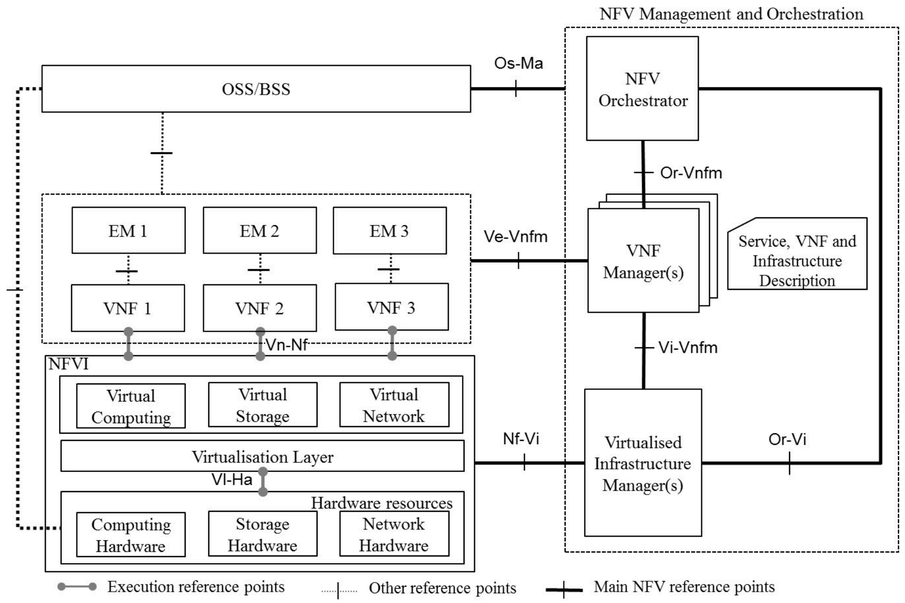
Liang Geng, Mingui Zhang, Mike McBride, Bing Liu; **Problem Statement of Edge Computing beyond Access Network for Industrial IoT**; draft-geng-iiot-edge-computing-problem-statement-00; IRTF T2TRG Meeting at IETF-100; Singapore; November 2017

Data Logistics



Also cf. Srikathyayani Srikanteswara, Jeff Foerster, Eve Schooler:
ICN-WEN Information Centric-Networking in Wireless Edge Networks;
Presentation at ICNRG@IETF-98, March 2017
https://www.ietf.org/proceedings/98/slides/slides-98-icnrg-information-centric-networking-in-wireless-edge-networks-eve-schooler-00.pdf

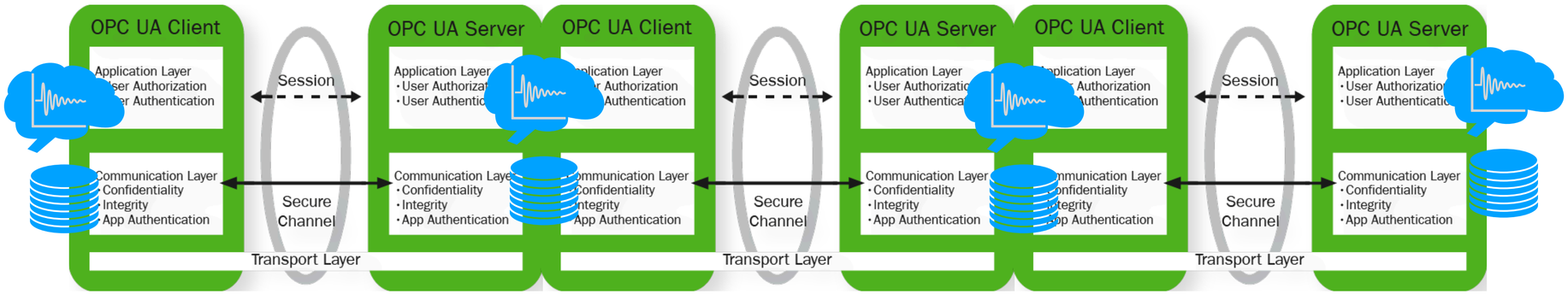
Different Perspectives on Compute & Networking



(Virtualized) Compute Servers in Networks

Networked Computations

In-Network Computing With Client-Server Protocols



- Overlays
  - Connection-based security
  - Client-server / broker-based
- Limited Scalability
  - Pub-sub distribution to many clients through single-server bottleneck
- Limited efficiency
  - Cannot share data directly
- Limited performance and robustness
  - Network cannot assist data dissemination

Adding a little computation to a data kiosk system is not exactly distributed computing.

What If…

We leveraged modern technology instead?

- **Unikernels, Super-light-weight-VMs**
  - Computation not as a static service but as a dynamic capability
- **Light-weight scripting**
  - 


JerryScript: JavaScript engine for the Internet of Things
- **Trusted Execution Environments**
- **Data-oriented communication and programming abstractions**
  - Information-Centric Networking, Named Function Networking
  - Reactive Programming
- **Decentralized Trust Management**
  - Distributed Consensus Protocols for Infrastructure Services
  - Finding stuff, trusting things, nano-payments

What If...

We leveraged modern technology instead?

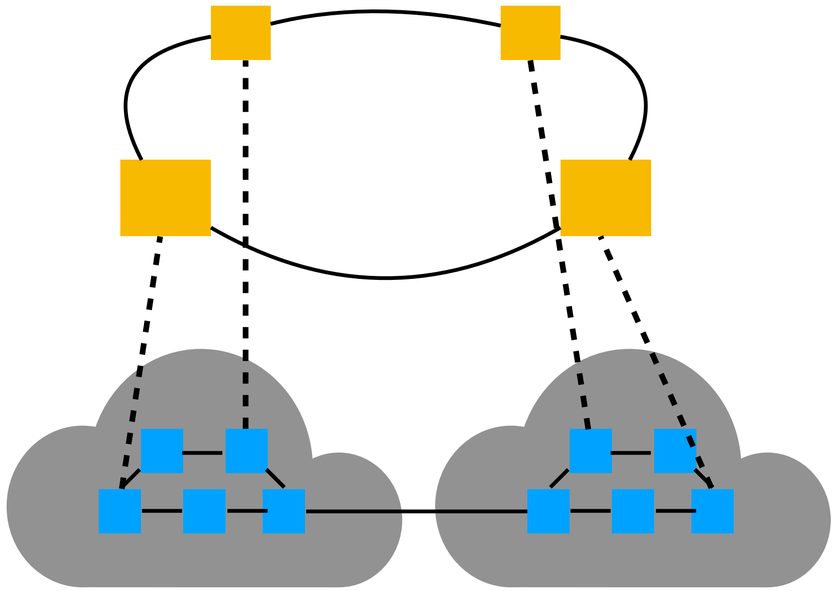
Towards building networks where computation is a first-order service — not an afterthought

Vision:
Compute-First Networking

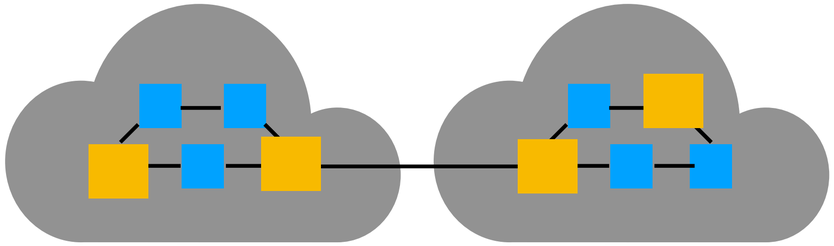
- Leverage increasing availability of computation at different scales
- Distributed computing as first-order principle — not as an afterthought through overlays
- Enable a wide range of new applications that are not possible or not easy to realize today
- Create general principles and architectures that can be mapped to different environments: edge analytics, DC Big Data processing, in-network computing in access networks etc.
- Fundamentally change the way we perceive and use ICT


路漫漫其修远兮
吾将谨慎而行之
The road is long and hard. I shall be careful with my steps.

Overlays vs. Networked Computing



- Decoupling higher layer functionality from lower layer network
- Simple network layer — intelligence in the overlay
- Generality over efficiency

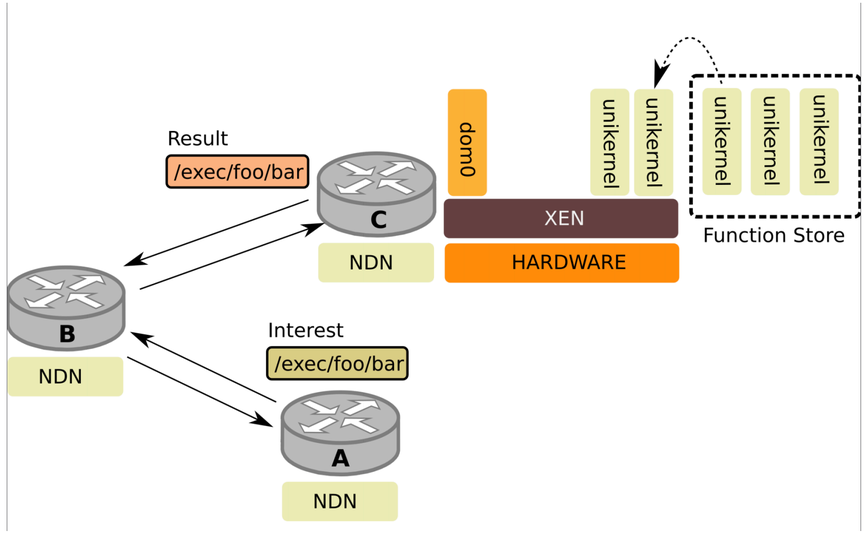


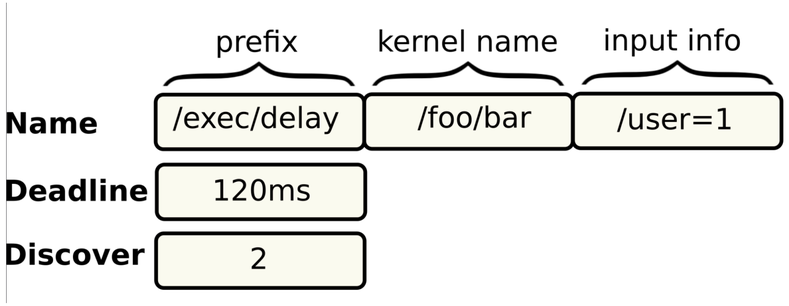
- Integrating functionality into network data plane
- Closing “gap” between applications and networks
- Efficiency over generality?

Compute-First Networking: Principles and Features

- **Distributed Computing Framework**
  - Conceiving networking and computation holistically
  - Concept of “data plane” for distributed computing: self-organized, self-optimizing, networking with as little management/orchestration as possible
  - Applicable to many current and future use cases
- **Principles**
  - Connection-less communication with a strong security model
  - Computation as a first order principle
  - Application-agnostic platform
  - Multi-tenancy as a first-order principle
- **Key features**
  - Highly dynamic
  - Agnostic to (access) network technologies
  - Agnostic to specific virtualization technologies (compute can run on different platforms)
  - Natural APIs to applications
  - Works well in well-connected (e.g., cloud-based) scenarios, without depending on cloud: decentralized operation possible

Named Function as a Service (NFaaS)





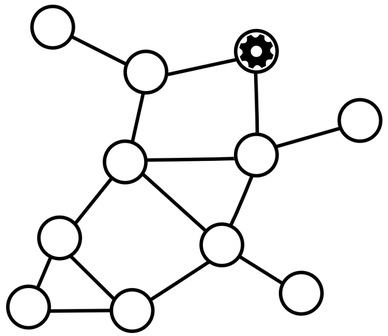
NFaaS

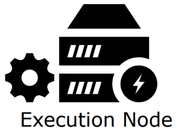
- Completely distributed
- Moving function where they're needed
- Functions as stateless unikernels
- Nodes run popularity contest
- Different forwarding strategies

Michal, Krol, Ioannis Psaras; **NFaaS: Named Function as a Service**; ACM ICN 2017

Decentralized Computations

Named Function as a Service





SPOC

- Automatic payments and result verification
- Based on Smart Contracts and Intel SGX
- No 3rd parties involved

- Secure against Rational Attacker
- Minimal computational overhead
- No calls privacy

Michał Król , Ioannis Psaras; **Decentralized Computations**;
Presentation at IRTF Proposed DINRG Interim Meeting; February 2018

Trust Management

- **Assumption: CFN will enable a rich eco system of distributed applications**
  - Very large number of application modules / compute functions in the network
  - Security will be a major challenge: Distributed systems tend to enlarge attack surfaces: many components instead of one
  - Especially trust management and authorization: Who is allowed to access which function, and how can you trust identities?
  - Authorizing network/compute usage
  - Authorizing access to shared data and computation results
  - Trusting compute functions and execution platforms
  - Need to automate verification and enforcement of security policies in the network
- **Decentralized Trust**
  - Enabling nodes, networks, organisations to trust each other
  - Without relying on centralized trust infrastructure

Food for Thought

- Lack of accepted common basis for edge, fog, in-network computing seems to suggest need for principled approach: **Compute-First Networking**
- What is a good balance between generality and efficiency?
- To what extent can/should we empower the data plane?
- What are the requirements for **Decentralized Trust Management**? (DINRG meeting Friday morning)

Semantics for Security
Security for Semantics

Carsten Bormann 2018-07-19 T2TRG

(1) Security Information has Semantics

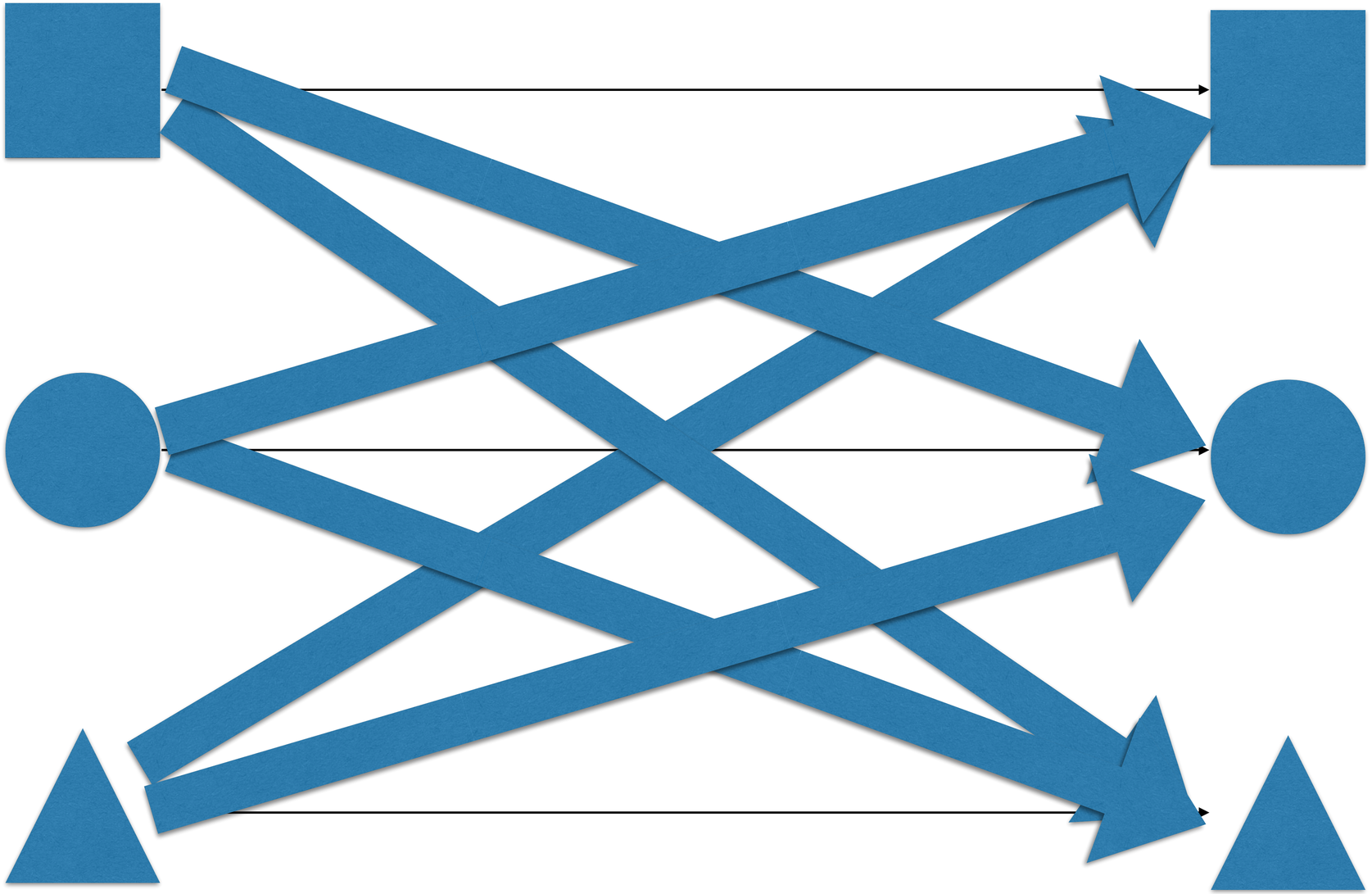
- Access Control data (ACLs, capabilities, tokens)
- Identities, Claims, Assertions, Attestations
- Privacy labels, privacy preferences

Heterogeneous Security Environments

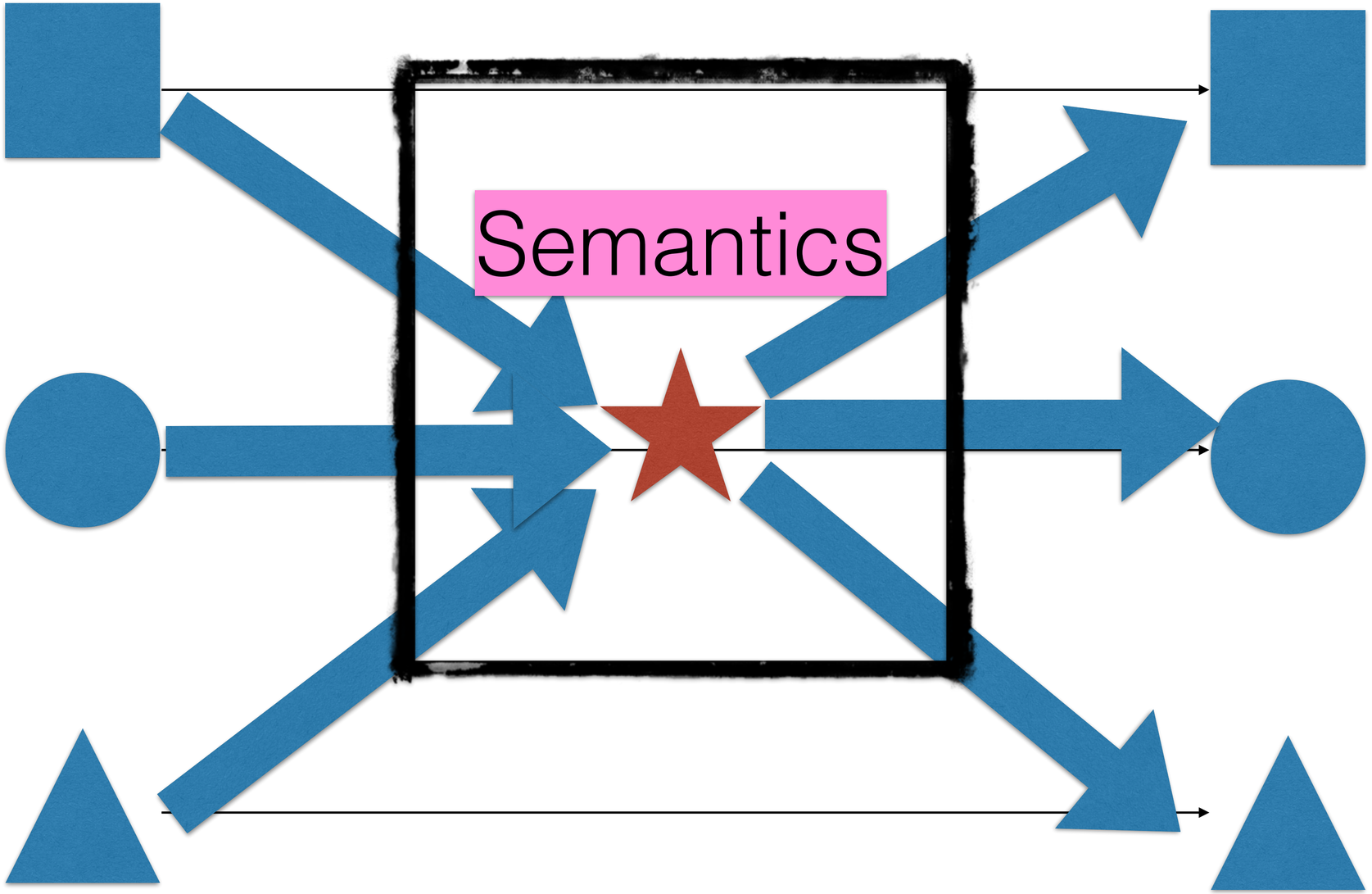
- Might need to make decision based on security data from different ecosystems (e.g., use identity from ES1 to access resources structured along ES2)

- Ecosystems often have tacit security properties
  ➔ heterogeneity generally requires making those explicit

$n^2 - n$



2n



How secure is semantic processing of security data?

- Even with tacit parts exposed: Fuzzy parts of the inference chain are too easy to attack
- Is there a way to apply abstracted vocabularies that continues to generate provable security properties?

(2) Semantic information needs security

- Semantic data needs confidentiality and integrity
- How discoverable do you want to be?
- How can other systems interact with you without discoverability?
- Highly relevant for security semantics, but for all other kinds of semantics, too

Make provenance and authorization part of inference

- Results of inferences are generally only as trustworthy as the combination of trustworthiness of the inputs
- Need to preserve provenance of all data that go into inference
- Need to apply authorization calculus to all inferences — from fact-based to claim-based inference

Security semantics of inference results

- What are the disclosure/privacy requirements on data generated from inferences?
- Labeling with provenance may already disclose too much
- Hiding behind indirections can help (and simplify!)

Getting the communities to talk

- Security community ≠
  Semantics community

- The problems are not new, but:

  - IoT often has complex multi-stakeholder security objectives

  - IoT needs semantic technology for wide-scale integration

Wrap-up