T2TRG: Thing-to-Thing Research Group

IETF #96 summary meeting
July 19th 2016, Berlin, Germany

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/2016-ietf96**

Agenda

- 16:20 (Chairs) RG status update
- 16:30 (Chairs) Summary from RIOT Summit
- 16:45 Hannes, Stephen, Carsten:
  Summary from IOTSU IAB Workshop
- 17:15 Matthias Kovatsch:
  Update from W3C WoT IG and WG
- 17:35 (Authors) T2TRG documents
- 17:50 Tibor Pardi:
  Secure, decentralized, blockchain based IoT (talk)
- 18:10 (Chairs) Future activities

Agenda

- 16:20 (Chairs) RG status update
- 16:30 (Chairs) Summary from RIOT Summit
- 16:45 Hannes, Stephen, Carsten:
  Summary from IOTSU IAB Workshop
- 17:15 Matthias Kovatsch:
  Update from W3C WoT IG and WG
- 17:35 (Authors) T2TRG documents
- 17:50 Tibor Pardi:
  Secure, decentralized, blockchain based IoT (talk)
- 18:10 (Chairs) Future activities

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

Done so far

- Chartered in December 2015. Multiple meetings before official chartering co-located with IETF meetings and with W3C Web of Things (WoT) group
- 2016: RG meeting at Nice co-located with W3C WoT, at San Jose co-located with IAB IoT**SI** WS, at Buenos Aires with the IETF meeting; participated in Dublin IAB IoT**SU** WS
- Three RG deliverable documents in progress on REST and security; multiple new documents on REST interaction
  ➔ later today
- Outreach (e.g., organizations like OCF and Bluetooth SIG)

Where are we going

- Work on RG deliverables and outreach continues
- Future meetings co-located with good research venues (2017)
- Meetings co-located with open source activity
    - RIOT summit right before this meeting
    - Eclipse IoT meeting (October in Southern Germany? **TBD**)
- Benchmark/reference scenarios
    - Initial discussion in various drafts and slides
    - More elaborate documentation by end of 2016

Agenda

- 16:20 (Chairs) RG status update
- 16:30 (Chairs) Summary from RIOT Summit
- 16:45 Hannes, Stephen, Carsten:
  Summary from IOTSU IAB Workshop
- 17:15 Matthias Kovatsch:
  Update from W3C WoT IG and WG
- 17:35 (Authors) T2TRG documents
- 17:50 Tibor Pardi:
  Secure, decentralized, blockchain based IoT (talk)
- 18:10 (Chairs) Future activities

RIOT Summit
July 15 - 16, 2016

http://summit.riot.org

In Berlin, days before IETF96

★ bringing together RIOTers, beginners & experts

★ gathering people interested in the IoT in general

★ plenary talks, hands-on tutorials & demos

RIOT Summit 2016

- ~ 135 developers and researchers met in Berlin
- RIOT = Research operating system for IoT
  (microkernel-based, full-fledged network stack)
  Addressing “M-class” platforms (microcontrollers)
  Can make good use of modern CPUs (32 bit)
  Has 6LoWPAN, CoAP, CBOR, …
- Half a day for breakout groups
  T2TRG: “The Web & the IoT: Design, Hacking, and Discussions”
  - Learning about implementation approaches and experience with relevant protocols

General issues

- What should be part of a “starter pack” for IoT developers?
  (potential for I-D about basic setup of an IoT node)
- What have we learned about memory management in constrained devices (≠ malloc())?
  - Constant tension between
    - optimizing for constrained devices
    - code-reuse for “A-class” platforms (Linux etc.)
    - ability to merge in open-source contributions

CoAP implementation

- One size does not fit all
  - from pure protocol parsers to highly flexible libraries
  - discussed microcoap, libcoap, and new gcoap
    - Also: Cloud-/Hub-side (e.g., aiocoap)
    - Limited experience with resource-directory implementations

Hypermedia Controls, W3C Web of Things

- New JavaScript engine JerryScript, fits upper M-class (using 1024 KiB/128 KiB as a reference platform)
  - One target for mobile code (but don’t ignore Lua)
- Discussion of the different roles different classes of devices can take in the W3C Thing Description approach

Data formats

- Floating point is still costly (SenML!)
- JSON libraries are larger than one thinks (printf!)
    - Several “M-class” CBOR libraries now available (RIOT’s CBOR, cn-cbor, tinycbor)
- Implementation experience with SenML (feedback mostly a need for clarifications)

Security

- TinyDTLS (Eclipse) as a reference platform
  - Good experience with focused set of cipher suites (PSK)
  - Somewhat chaotic advances in crypto providers, moving target
  - Complement DTLS with object security (COSE)
- random number generators: entropy pools
- Discussion of OTA needs to address OS-specific as well as security-related issues

Next Steps

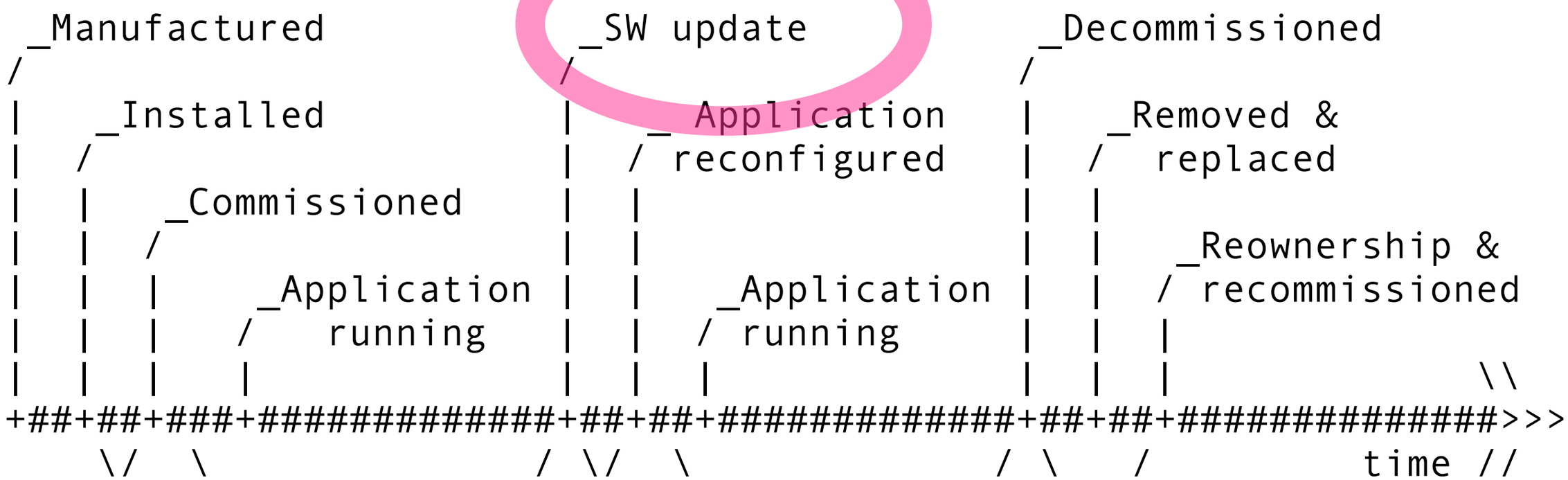
- Session was generally regarded as useful
- Follow-up:
  - Join in via the periodic online meetups
  - Transfer information between RIOT and IETF/IRTF lists

Agenda

- 16:20 (Chairs) RG status update
- 16:30 (Chairs) Summary from RIOT Summit
- 16:45 Hannes, Stephen, Carsten:
  Summary from IOTSU IAB Workshop
- 17:15 Matthias Kovatsch:
  Update from W3C WoT IG and WG
- 17:35 (Authors) T2TRG documents
- 17:50 Tibor Pardi:
  Secure, decentralized, blockchain based IoT (talk)
- 18:10 (Chairs) Future activities

SOLACE: Smart Object Lifecycle Architecture

- Processes for **usably secure** lifecycle (changes of ownership, authorization, privacy, …)

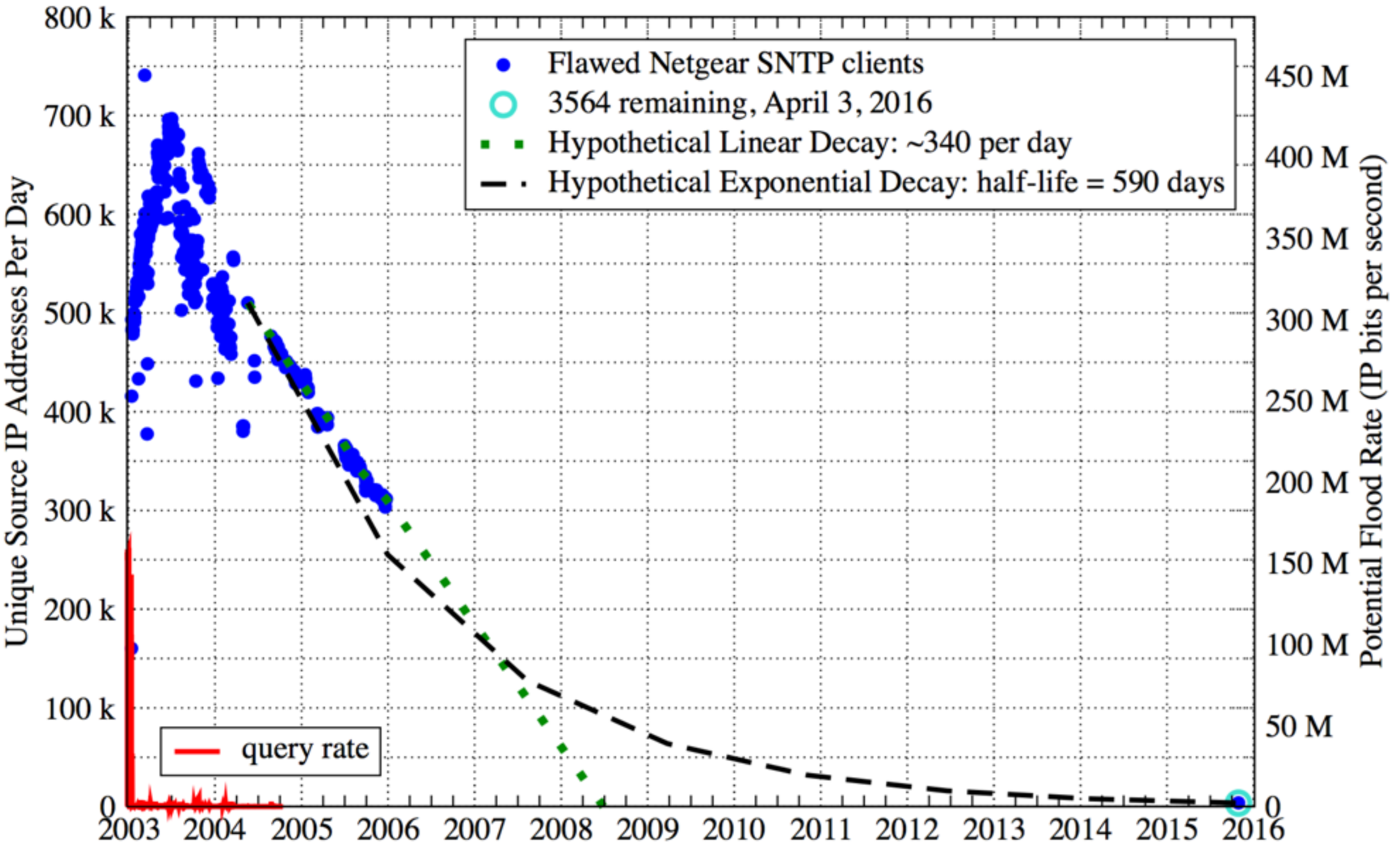


The lifecycle of a thing in the Internet of Things

[draft-garcia-core-security]

Internet of Things Software Update Workshop (IoTSU)

Dublin, 2016-06-13/-14


Flawed Netgear SNTP clients
3564 remaining, April 3, 2016
Hypothetical Linear Decay: ~340 per day
Hypothetical Exponential Decay: half-life = 590 days
query rate
Unique Source IP Addresses Per Day
Potential Flood Rate (IP bits per second)
800 k
700 k
600 k
500 k
400 k
300 k
200 k
100 k
0
450 M
400 M
350 M
300 M
250 M
200 M
150 M
100 M
50 M
0
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016

[Plonka]

Nest owners left in the cold: Bug forces smart thermostats offline as frigid temperatures arrive

- **It took two weeks from the update to when the bug caused software issues**
- **Nest has confirmed 99.5% of all devices are back online**
- **For those still having issues, there is a nine-point plan on the Nest website**
- **Short-term fix is recharging the device by plugging USB cable in the port**

By Stacy Liberatore For Dailymail.com

PUBLISHED: 21:20 GMT, 14 January 2016 | **UPDATED:** 02:43 GMT, 15 January 2016

A software bug has hit tons of Nest smart thermostats, draining the batteries and knocked heating systems offline - just as the polar vortex arrives.

Nest owners have reported waking up to frigid homes over the past few days.

The firm stated the issue stems from an update that was released last month - but has not yet been able to fix it.

'We had a bug that was introduced in the software update that didn't show up for about two weeks,' Nest co-founder Matt Rogers told the **New York Times**.

This fiasco came from the update version 5.1.3, which hit homes last month.

'Woke up to a dead nest and a very cold house,' a commenter wrote on the company's forum.

'Not good when you have a baby sleeping!'

Nest believes they have fixed 99.5 percent of all affected customers, according to **Engadget**, but if you fall into the .5 percent, the company has a nine-step plan on the website to get the device working again.

The **Nest Support Page** describes what to look for to confirm your system has been affected such as, a message on the display that reads, 'Please remove the thermostat from its base, then reattach it' or the device's animations are slower than usual.

HOW TO TELL IF YOUR NEST HAS THE BUG

Your thermostat is offline in the Nest app and disconnected from Wi-Fi.

Your thermostat tells you that its battery is low and it needs to shut down.

Your thermostat's animations are noticeably slower than usual.

Your thermostat shows you a message: 'Please remove the thermostat from its base, then reattach it.'

Your thermostat's display is dark and it's unresponsive. You may also see a blinking red or green light above the display.

Your thermostat can't control your heating or cooling system.

'In some cases, this may cause the device to respond slowly or become unresponsive.'

The firm stated the issue stems from an update that was released last month. Nest believes they have fixed 99.5 percent of all affected customers, according to Engadget , but if you fall into the .5 percent, the company has a nine-step plan on the website to get the device working again

Software update causes $286 million Japanese satellite to break apart in orbit

By Lee Mathews May. 10, 2016 11:15 am

MORE GEEK

NASA finds a mineral on Mars that could rewrite its entire history

You've probably experienced a bad software update before. Maybe it slowed down your old iPhone. Maybe it reduced your laptop's battery life. It probably didn't destroy your $286 million satellite.

That is, unfortunately, exactly what happened to Japan's space agency JAXA

Internet of Things Software Update Workshop (IoTSU)

Session I - experiences

Overview

“Cortex M Class” Type of Device

- Hardware offers basic isolation features (e.g., MPU)
- Often do not run an operating system (bare metal).
- May run a RTOS
- Single firmware image / MCU
- Firmware image comes from OEM (but may contain libraries
- Product may contain multiple MCU

“Cortex A Class” Type of Device

- Hardware offers hardware isolation features (e.g., MMU, virtualization capabilities)
- Run standard OS (e.g., Linux)
- Software updates use sophisticated package managers
- Software comes from various sources.
- Hardware may come with a trusted execution environment (TEE).

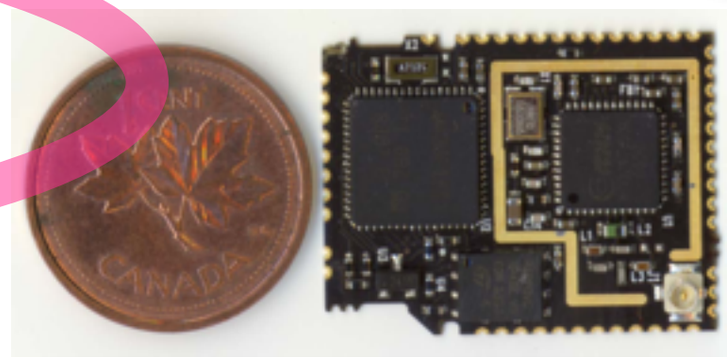
RFC 7228:
(Class-0)
Class-1
Class-2

Constrained nodes: orders of magnitude

10/100 vs. 50/250

- There is not just a single class of “constrained node”
- Class 0: too small to securely run on the Internet
  - “too constrained”
- Class 1: ~10 KiB data, ~100 KiB code
  - “quite constrained”, “10/100”
- Class 2: ~50 KiB data, ~250 KiB code
  - “not so constrained”, “50/250”

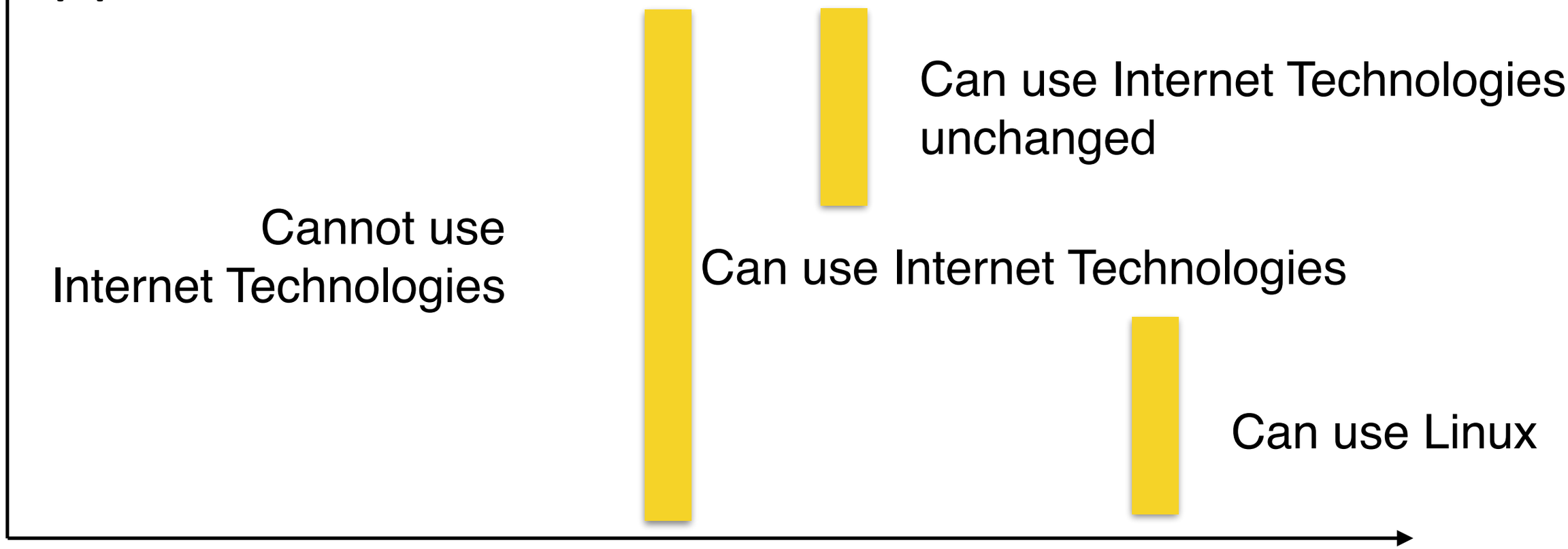
RFC 7228



- These classes are not clear-cut, but may structure the discussion and help avoid talking at cross-purposes

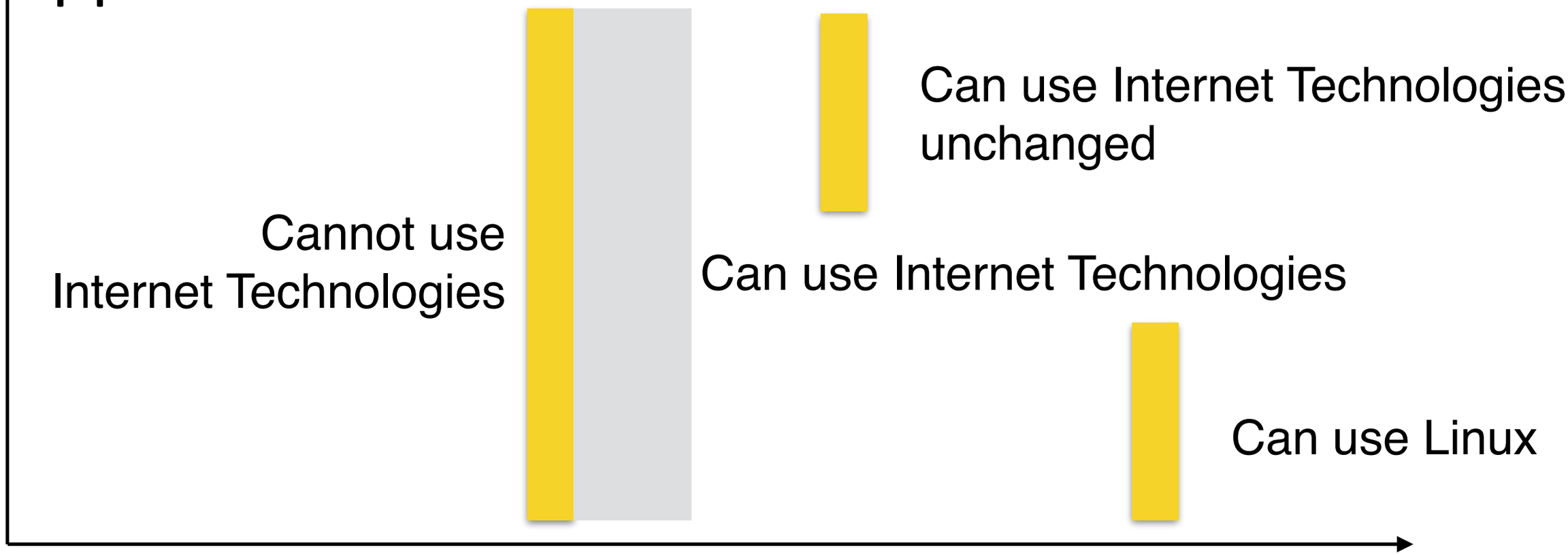
Moving the boundaries

- Enable Internet Technologies for mass-market applications

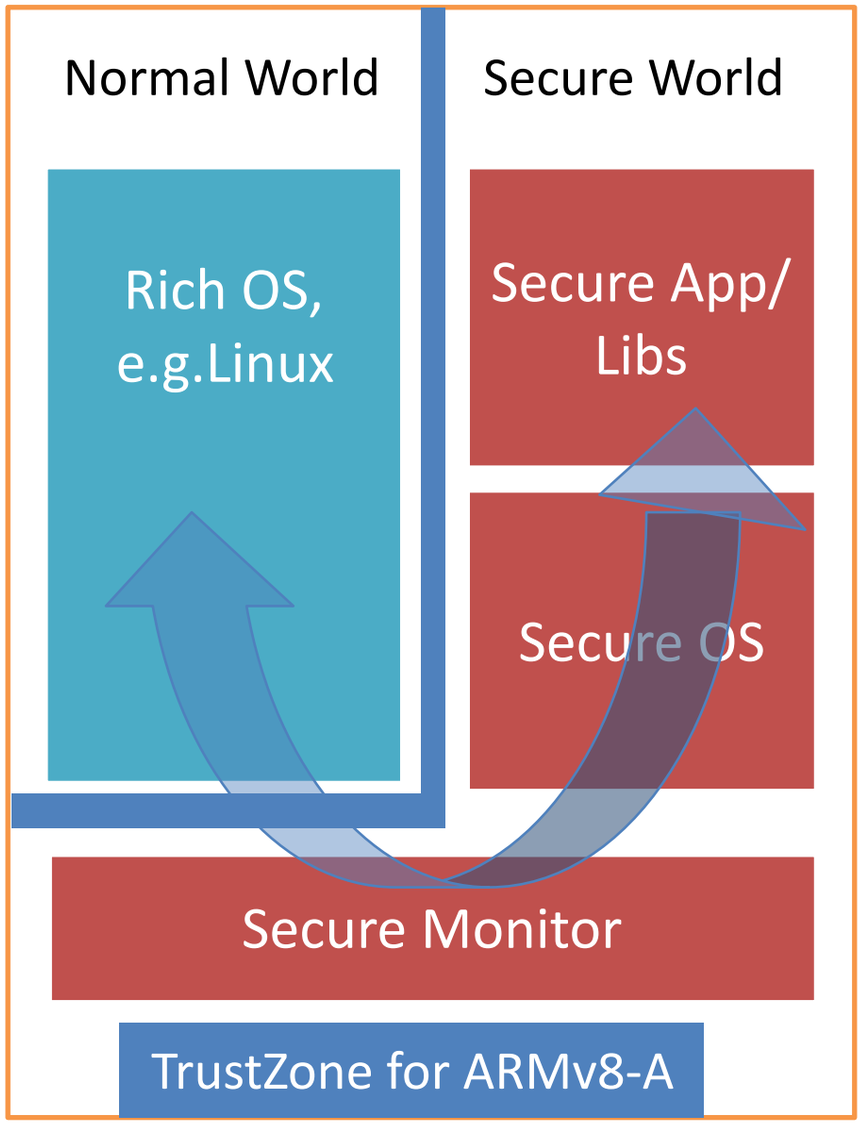


Moving the boundaries

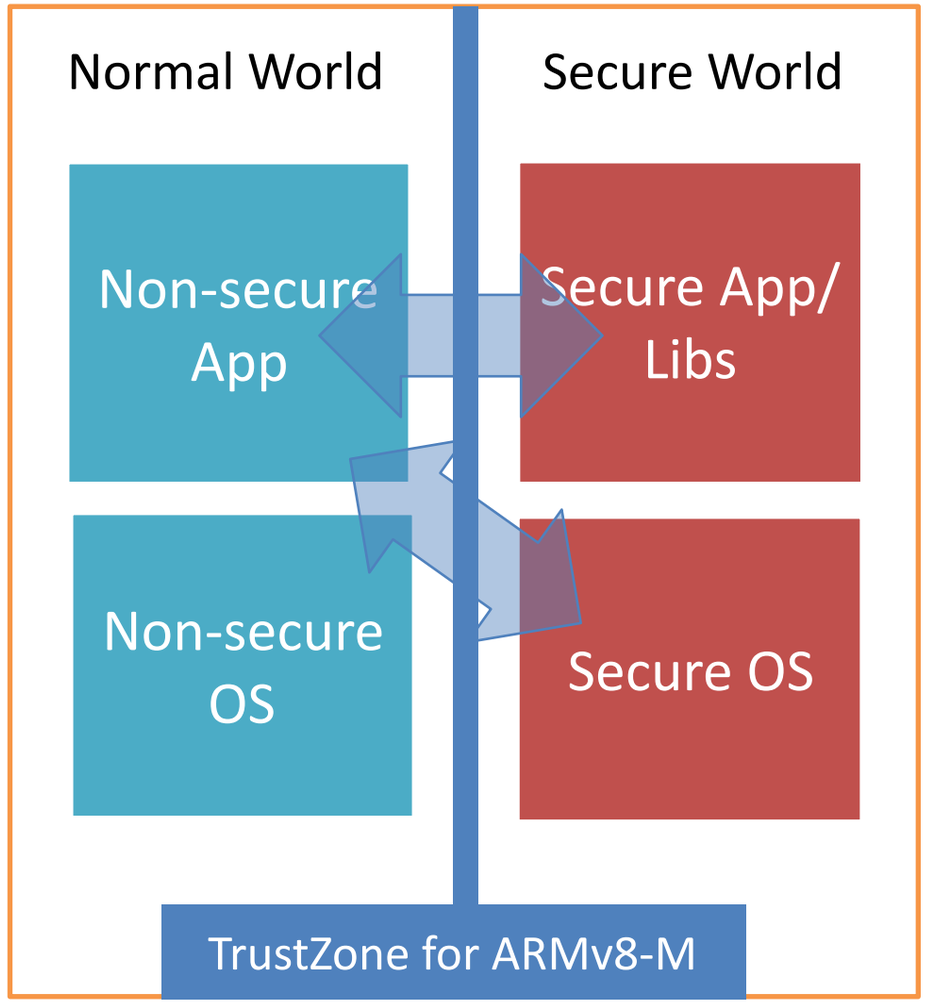
- Enable Internet Technologies for mass-market applications



TrustZone for ARMv8-A and ARMv8-M



Two separate software update mechanisms; one for normal world and one for the secure world.



Single software update mechanism? Maybe different developer experience.

Why are these features there?

- Because security is good? Nah.
- Devices with DRM (set-top boxes)
- → Features that go against the wishes of the device owners!

Intel IoT SoCs

Ned Smith
IoTSU Workshop
June 2016

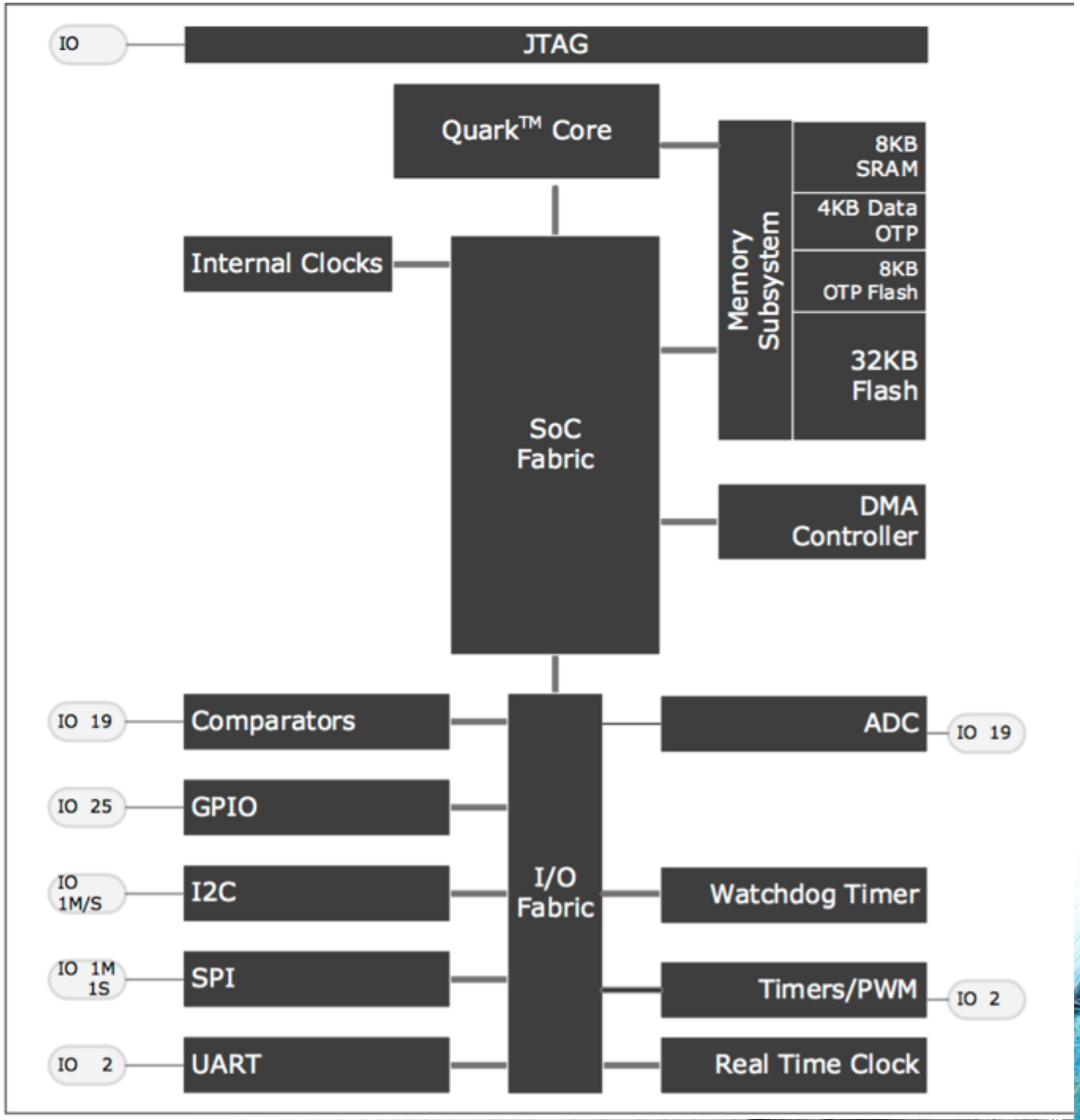
Intel Quark and Atom for IoT

- Quark D2000 SoC
  - MCU
    - 32-bit x86
    - 32 MHz (settable to 4/8/16 MHz)
    - APIC w/ 1 32-bit core timer
  - Memory
    - 32K Flash (4 protection ranges)
    - 8K SRAM (4 protection ranges)
    - 8K OTP RAM (code)
    - 4K OTP RAM (data)
    - MMU
  - Other
    - 2 32-bit timers / PWM
    - Always on counter
    - Always on timer w/ wake
    - Watchdog timer
    - <3.5uA - <30mA
  - Future
    - EPID
- Atom E3800 SoC
  - CPU
    - 64/32-bit x86 (1,2,4 cores)
    - 1.3 – 1.9 GHz
    - 32K L1, 1M L2 cache
  - Memory
    - DDR3 X 2
    - MMU
  - Security
    - DRNG
    - VT-x
    - AESNI
    - 128-bit carryless mult
    - Secure boot
  - Other
    - Timers
    - <100mW – (3 – 10 W)
  - Future
    - EPID

Updatable Components

- Quark D2000 SoC
  - uCode
  - BIOS
  - Option ROMs(?)
  - Protection ranges (4)
    - System image(s), Secure storage, BIOS
  - OTP RAM
    - First use
- Atom E3800 SoC
  - uCode
  - BIOS
  - Option ROMs
  - Hypervisor
  - Guest OS(s)
  - Frameworks
  - Apps
  - Secure boot
    - First use

Quark D2000 SoC Layout



Relevant Papers

- Paper 01: Housley, Position Paper for Internet of Things Software Update Workshop (IoTSU)
- Paper 10: Thomas, Incentivising software updates
- Paper 15: Zappaterra, Software Updates for Wireless Connected Lighting Systems: requirements, challenges and recommendations
- Paper 21: Zugenmaier, Updates in IoT are more than just one iota
- Paper 25: Plonka, The Internet of Things Old and Unmanaged
- Paper: Tschofenig, Software and Firmware Updates with the OMA LWM2M Protocol
- Jimenez and Ocak, Software Update Experiences for IoT

Incentives

- Paper 10: Thomas, Incentivising software updates
- Paper 25: Plonka, The Internet of Things Old and Unmanaged

- Companies often fail to ship software updates. Why? Can we so something about it?
- Question: Can we monitor the performance of different companies at supplying software updates to their customers?

Types of Devices

- “Jellybean” vs. regulated (e.g., healthcare)
- Security impact (door lock)
- Safety impact (e.g., Nest!)
- Pet vs. cattle

The role of the user

- Users don’t want upgrades
  - “It works well enough as it is”
  - Evil Deviceco might be deleting features I rely upon
    - or bugs I rely upon (!)
    - → rollback !?
- A single upgrade going bad can be closing the window for a long time

iOS upgrade statistics

- Looks great
- But then:
  - high device churn
  - lots of nagging by iOS
  - “pet” status
  - dependency of new apps on OS upgrades

All Platforms:

9.X	85.9%
8.X	7.1%
7.X	4.1%
6.X	2.0%
5.X	0.8%
4.X	0.1%

Last Updated: Jun 21, 2016 07:30:54
https://david-smith.org/iosversionstats/

Security

- Paper 01: Housley, Position Paper for Internet of Things Software Update Workshop (IoTSU)
  - Is about securing firmware packages.
  - Russ: Features of RFC 4108 and design rational.
  - Question: What features could be added (Merkle Tree Signatures)?

Internet of Things Software Update Workshop

Session II - Requirements and Constraints

Session Leader: Russ Housley

Topics from the Position Papers

- Device Requirements
- Infrastructure Requirements
- Manufacturing Requirements

- Questions that were raised that might reveal some other requirements

Device Requirements

- Not limited to full firmware update
- Provide compatible firmware for various components within the device
- Support devices with multiple owners
- Different authorities may update software for different parts of the device
- Identify dependencies among various software updates
- Digital signature and encryption on the update
- Allow multiple signatures on the update
- Minimize device downtime due to update processing
- Recovery procedure when the device gets hacked
- Support over-the-air software update, probably requires polling

Infrastructure Requirements

- Support many different approaches to digital signatures
- One infrastructure can support open- and closed- source
- One device can act a local server for neighbors
- Perform some digital signature checks on behalf of the served devices, such as revocation checking
- Multicast the same updates to many similar devices
- Hide complexity associated with NATs and Firewalls from the devices

Manufacturing Requirements

- Fast and secure key generation

Questions from the Position Papers

- Can the device owner decide to accept/reject an update?
- Can we determine whether the update impacts other devices in the IoT?
- Can we handle end-of-service, end-of-feature, and end-of-device-support?
- Can a community take over support after the vendor decides to end-of-life a device?
- Can the user pick among updates when there is more than one available?
- Can we determine when a device is not active to apply the update?
- Can we do a better job preserving the privacy of the device owner?

Authentication (1)

- Can the firmware be trusted?
  - Can the **source** be trusted?
- Is it really for me?
  - Am I the right device for this FW? (HW revision!)
  - Do I have the other prerequisites (libraries, FPGA code, …) or do they need to be upgraded in sync?
  - Is the FW the right one for my usage situation? (Authorization!)

Authentication (2): Freshness

- Is the FW fresh?
  - downgrade attacks (revocation?)
    - version number comparison?
      - (but also prevents operational downgrades!)
  - weak upgrade attacks
  - sidegrade attacks?

Internet of Things Software Update Workshop (IoTSU)

Session V: Future Solutions

Transport

- Lighting industry with mesh networks (based on IEEE 802.15.4)
  - Paper 15: Zappaterra, Software Updates for Wireless Connected Lighting Systems: requirements, challenges and recommendations
- Low Power WANs
  - Paper 21: Zugenmaier, Updates in IoT are more than just one iota
- LWM2M
  - Tschofenig, Software and Firmware Updates with the OMA LWM2M Protocol
- Communication Patterns:
  - Jimenez and Ocak, Software Update Experiences for IoT
- Questions:
  - How to distributed firmware updates efficiently? How to reduce the amount of flash memory? What is the implication for security of image itself? How to avoid draining the battery?

Papers in this Slot

- Paper 03: Robert Bisewski, Comparative Analysis of Distributed Repository Update Methodology and How CoAP-like...
- Paper 05: Smith,Toward A Common Modeling Standard for Software Update and IoT Objects
- Paper 13: Schmidt,Secure Firmware Update Over the Air in the Internet of Things Focusing on Flexibility and Feasibility
- Paper 16: Adomnicai,How careful should we be when implementing cryptography for software update mechanisms in the IoT?
- Paper 20: Prevelakis,Controlling Change via Policy Contracts
- Paper 23: Birkholz,IoT Software Updates need Security Automation
- (but also see Paper 08, 11, …)

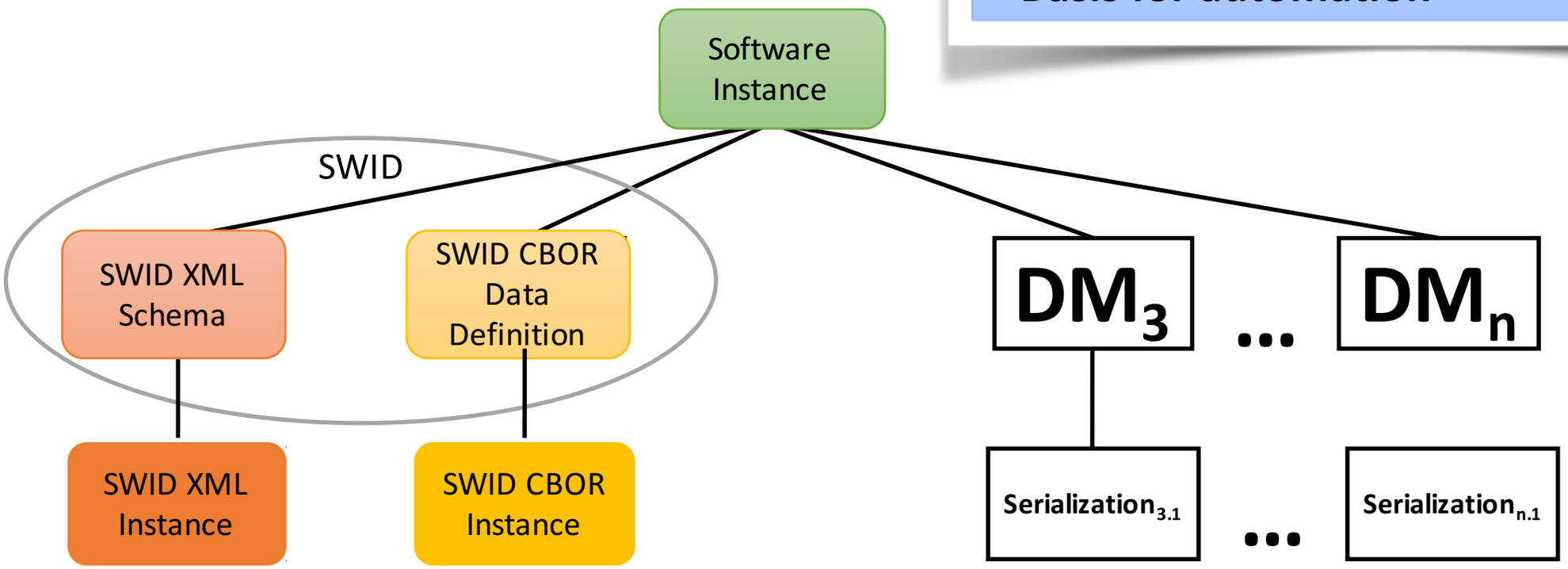
Updating a sea of devices

- What do I have
  - Device description (models, components — e.g., SWIDs)?
  - and can I trust what I believe (Attestation)?
- Push/Pull
  - Push: MPL and other multicast/flooding
  - (Pull: Doing proper congestion control)
- Limiting Damage
  - **Are we in Critical Operational State?**
  - Even better: Hitless Upgrades
  - Identifying dud upgrades, rollback

COSWID: Software-ID tags for constrained devices

IM, DM, and Serialization

- **Device describes itself**
  - **Can use hashes on device**
  - **Compare with source-based values**
- **Basis for automation**



TUDA: Time-based unidirectional attestation

- Remote Attestation: attempt to describe the integrity and trustworthiness of a host or device
  - Measurements of components (e.g., hash values)
- Protocols for RA typically bidirectional
  - Challenge for freshness
- TUDA: **Time-based unidirectional** attestation

#L2

Deployment Experiences & Issues
http://jaimejim.github.io/drafts/draft-jimenez-iotsu-soft-exp.txt

- Dealing with Sleepy endpoints: Caching is needed
- Device Initiated Communication: the common pattern we see from devices.
- Manager Initiated Communication: NATs make that very tricky -- COAP Proxy can be used
- Delegation on other nodes(GW): Very useful for some usecases
- Using Multiple Stacks: We have also seen that it is very common to have two stacks on devices, one for daily use and another for firmware upgrades, which is unrealistic on the constrained space.
- Runtime Discovery: A proposal on how software updates could be done with small upgrades- not once

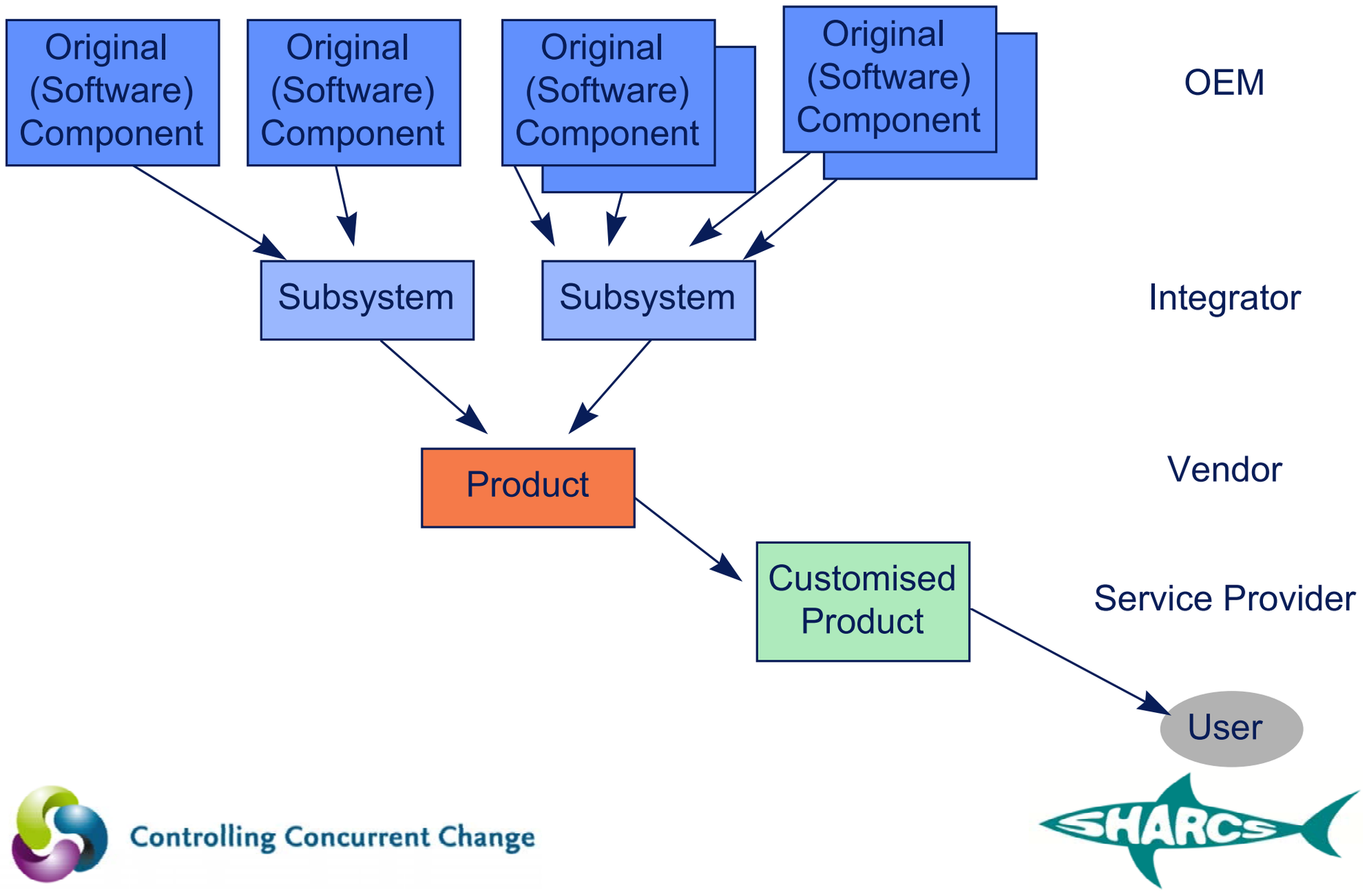
Technische Universität Braunschweig

Admission

- new software component “arrives”
- need to determine whether:
  - the new component is suitable for our system
  - the system can accommodate the new component
    - need to consider aspects such as:
      - services
      - load (memory, CPU)
      - interconnections (internal, platform, outside)
      - behavior

Technische Universität Braunschweig

Who do you trust?



Controlling Concurrent Change

SHARCS

Technische Universität Braunschweig

Policy Contracts

- credentials
  - X-509 certs with extensions
  - from one key to another key
  - attribute-based access control (ABACS)
- essentially say
  - this component **can do** this, this and this
  - and **needs** this, this and this resource/library/comm-channel etc.
- can enforce customization
  - e.g. integrator limits connectivity of component
- policy language can be “run” to determine access

Controlling Concurrent Change

SHARCS

Using RFC 2704 Keynote

```
if (
        (sensor == TRUE && switch == TRUE
                && memory >= 1000))
        || (sensor == TRUE && switch == FALSE
                && memory >= 700)
        || (sensor == FALSE && switch == TRUE
                && memory >= 350)
) -> TRUE
```

LICENSEE = ***integrator*** *public key*

AUTHORIZER = ***designer*** *public key*

signed_by = ***designer*** *private key*

Component-based approaches

- Components are important for
  - What do I have
  - Hitless upgrades
    - An ecosystem of upgrade sources
- Model the build process
  - Pre-built (possibly for a specific device)
  - Linking on device

IoT Object Modelling

IoT System Construction

#05

Information Model (IM)

Information Model Language (IML)
(e.g. OWL, RDF)

Data Model (DM)

Data Model Language (DML)
(e.g. JSON Schema, XML, RAML)

Object Serialization Method (OSM)

(e.g. CBOR, JSON, Base64)

Object Interaction Protocol (OIP)

(e.g. CoAP, HTTP)

Secure OSM

(e.g. COSE, JOSE)

Secure OIP

(e.g.TLS, DTLS, DBUS)

System Definition (SD)

System Definition Tools & Languages
(e.g. Yocto, Make)

Image Description (ID)

Packages, Bundles, Manifests, Versioning
(e.g. IPK, DEB, RPM, Clear)

Image Serialization

(e.g. ISO, CDROM, TAR)

Image Dependency Graph

(e.g. RPM)

Secure Image Serialization

(e.g. CMS, RPM)

Secure System Update

IoT Object Information Model (OIM)

Security Information Model (SIM)

Platform Image Information Model (PIM)

IoT Object Data Model (ODM)

Security Data Model (SDM)

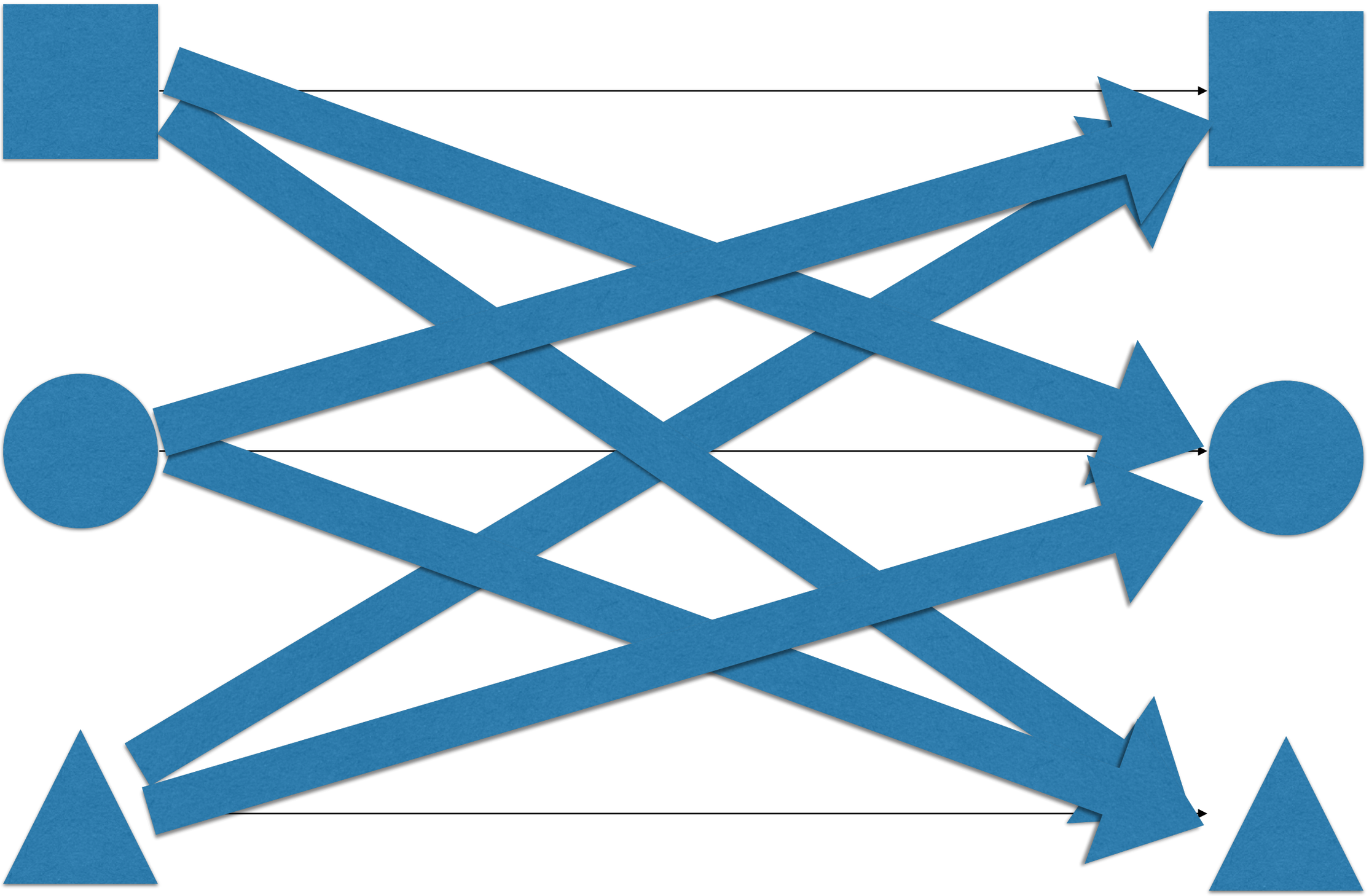
Platform Image Data Model (PDM)

IoT Common Object Serialization

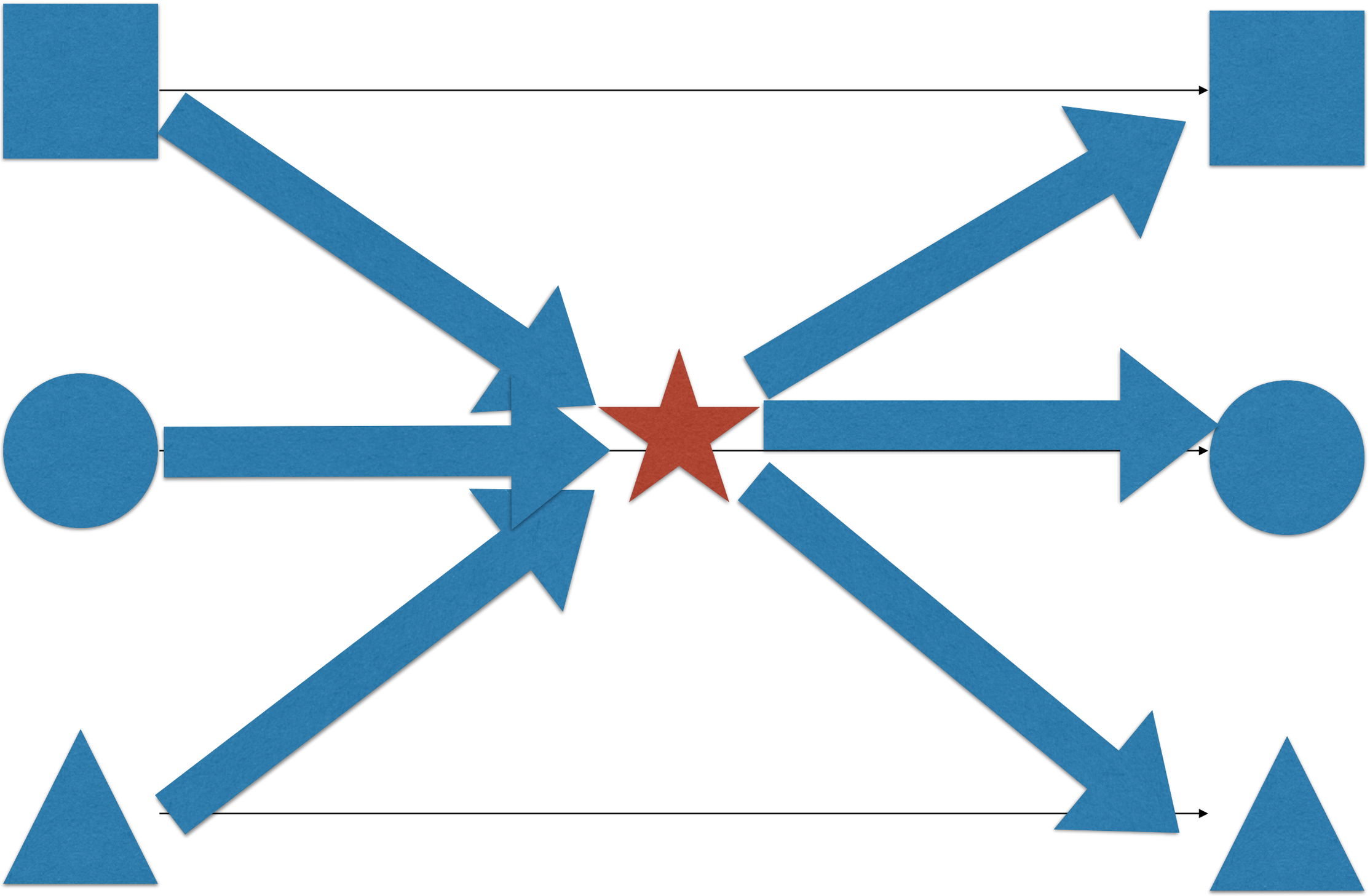
Mapping Data/ Information Models

IOTSI Workshop, 2016-03-17

$n^2 - n$



2n



1

2n

What is that hub?

Data loss?

②

Translating **data** between data models

vs.

Translating data **models**

③

Data/Information Models

vs.

Interaction Models

4

Semantic Level

Vocabulary

Taxonomy

Meaning

Ontology

Information Model

Data Model

Abstract Syntax

Serialization

Concrete Syntax

Marshaling Scheme

Message Transport Format

Encoding

⑤

How far can we get?

Limits to translation (e.g., security?)

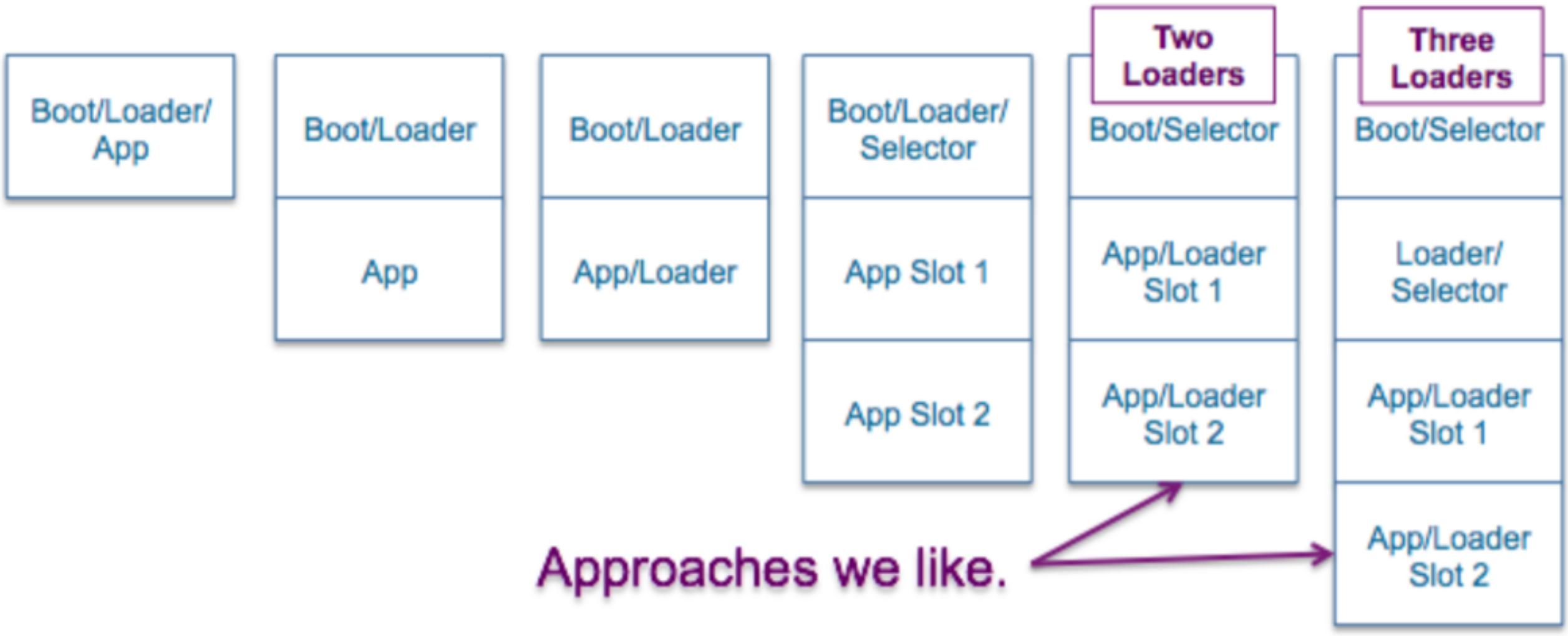
What is holding back components?

- do we know how to keep firmware componentized in **class-2 or even class-1** devices, or is this only for A-class devices?
- what are **safe update** procedures, in particular for class-2/class-1?
- how can we handle the issues that will prop up when various **versions** of various components meet each other as well as various **hardware revisions**? How can we use **modeling** to assess the security/safety issues of these combinations?
- what are the **non-technical** issues (**disclosure** of vendor relationships [Ted] and of "secret sauce" in general, **liability** considerations through a more complex set of combinations deployed and/or increased **hackability** of components, ...), and how can they be mitigated?

But then…

- There are systems that split ROM/flash
  - (Problem here: Flash part gets bigger each update as ROM code grows invalid)
- Some systems that provide hitless upgrade even upgrade config data and operational state

Evolving from…



Continuous Deployment?

IoT Software:
Towards Hardware Independence

- Need to evolve towards a state where 90% of the IoT software is hardware independent
- Else, we head to an Internet of buggy Things
- This is achievable with an efficient, open-source IoT software platform, e.g. RIOT

IoT Software: Components *vs* Full Firmware

- Open-source platform model for IoT software:
  - community maintains basic OS + network stack
  - vendors focus on small part of the software, e.g. application software, or low-level driver
- Bottom-line: different entity will update different parts of the software.
- Advantages: smaller software updates, end of vendor support does not necessarily imply end of security, vendor independent security maintenance…

RIOT Summit
July 15 - 16, 2016

http://summit.riot.org

In Berlin, days before IETF96

★ bringing together RIOTers, beginners & experts

★ gathering people interested in the IoT in general

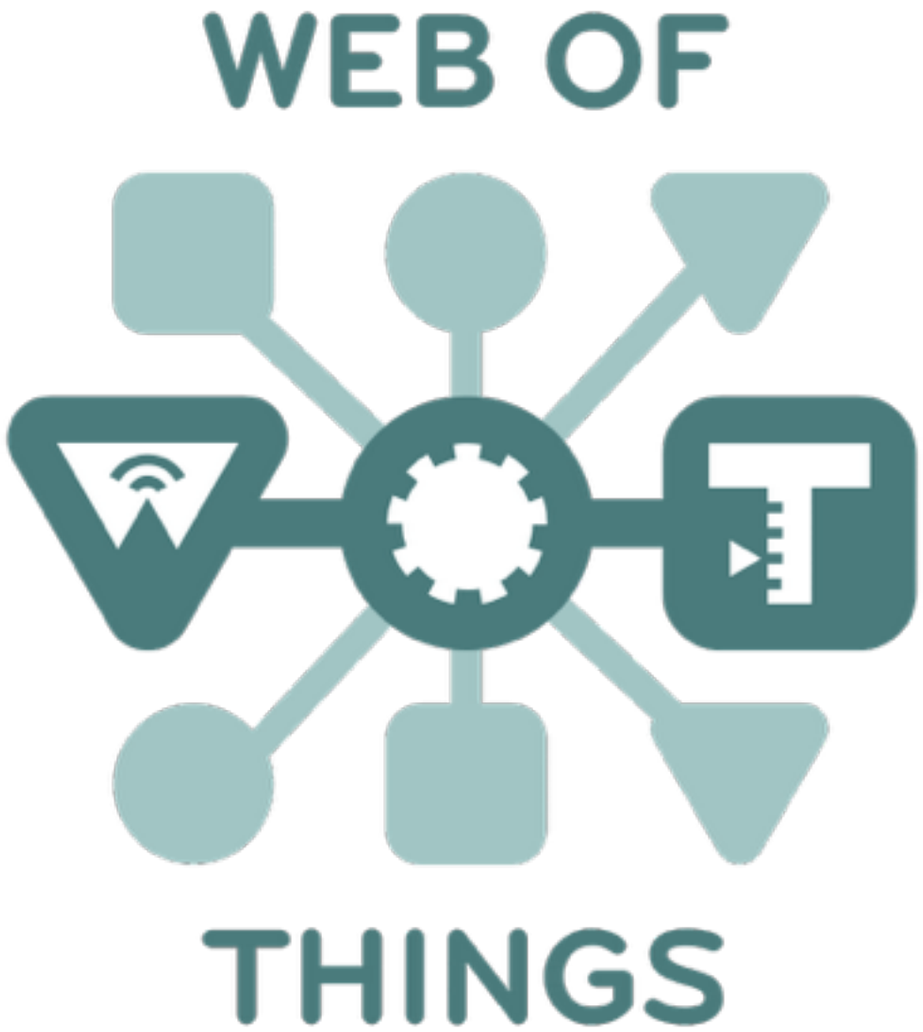
★ plenary talks, hands-on tutorials & demos

Übungsblatt

- **Aufgabe 1, 5 Punkte, Gruppe**
- Welche Internet-verbundenen (oder sonst vernetzten) Geräte besitzt/verantwortet Ihr? Findet jeweils heraus,
  - ob es Firmware-/Software-Updates dafür gibt
  - wo man die (autoritativ!) findet
  - welche Sicherheitsprobleme das Gerät hat und welche durch Updates gelöst wurden
  - evtl., wie gesichert der Update-Prozess ist
  - evtl., wie automatische Updates funktionieren
  - was eine guter Zeitpunkt für ein Update wäre, und wie das Gerät das evtl. herausfinden könnte
  - …
- Abgabe: Donnerstag, 30.06.2016 25:59 UTC

Agenda

- 16:20 (Chairs) RG status update
- 16:30 (Chairs) Summary from RIOT Summit
- 16:45 Hannes, Stephen, Carsten:
  Summary from IOTSU IAB Workshop
- 17:15 Matthias Kovatsch:
  Update from W3C WoT IG and WG
- 17:35 (Authors) T2TRG documents
- 17:50 Tibor Pardi:
  Secure, decentralized, blockchain based IoT (talk)
- 18:10 (Chairs) Future activities



T2TRG Summary

IETF 96, Berlin, Germany, 2016

http://w3c.github.io/wot/charters/wot-ig-2016.html

INTEREST GROUP RE-CHARTER

WoT Interest Group

- AC Review finished 15 July 2016
  - 34 support this Charter as is
  - 1 suggests changes, but supports the proposal
- IG Scope
  - Support proposed WG
  - Organize and run PlugFests
  - Collaborate with other SDOs, organizations, etc.
  - Investigate ideas for long-term goals

http://w3c.github.io/wot/charters/wot-wg-2016.html

WORKING GROUP CHARTER

Proposed WoT Working Group

- Roadmap
  - Integrate feedback from bilateral outreach
  - Resolution to submit on 27 July 2016
  - Start W3M Review period on 3 August 2016
  - Start AC Review period on 24 August 2016
  - Be able to start WG around October 2016
- Please have a look and send feedback
  - http://w3c.github.io/wot/charters/wot-wg-2016.html
  - Mailing list or GitHub Issues

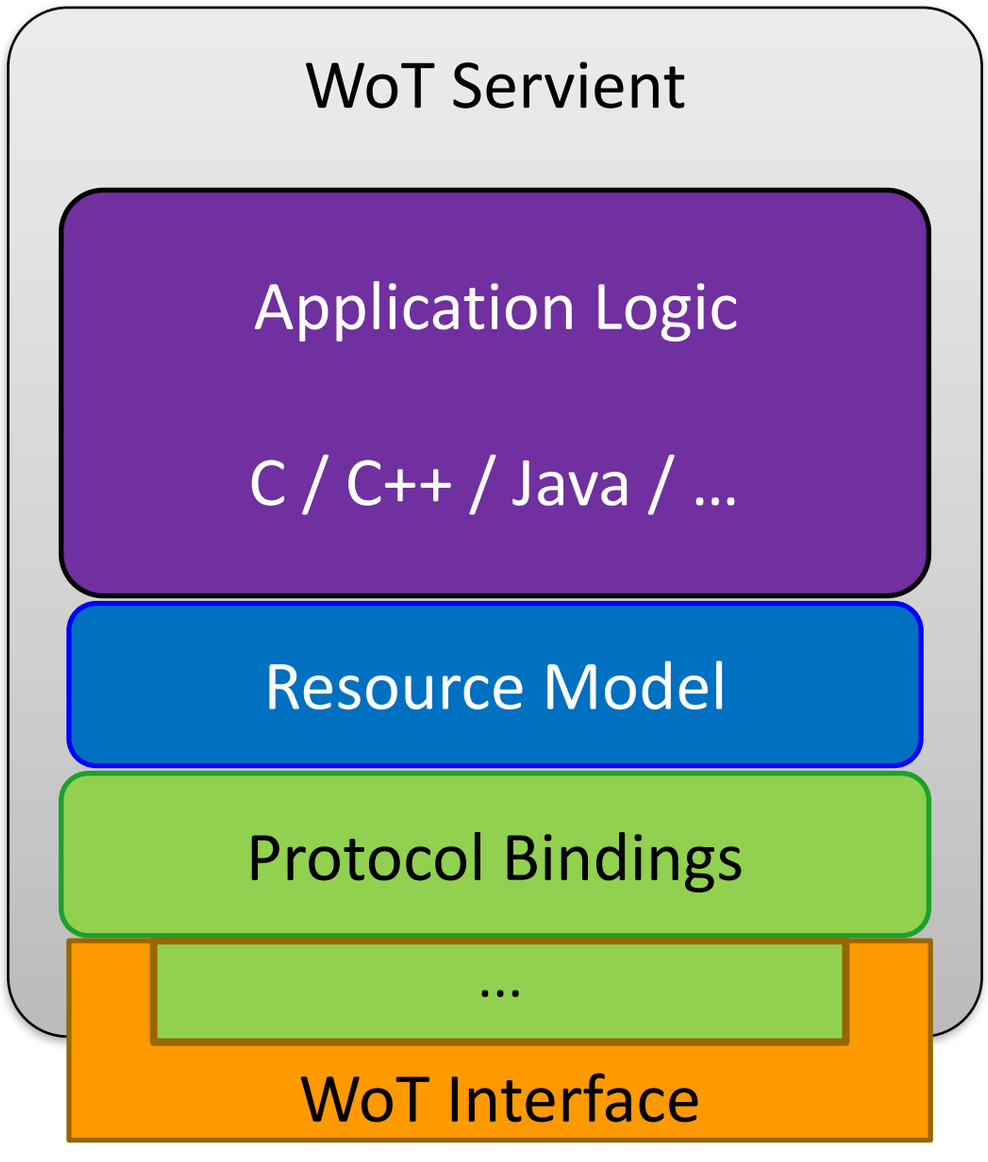
MAIN PROGRESS TOPICS

Thing Description (TD) Type System

- TD allows to plug in different systems
- Evaluation of popular type systems in Web apps
  - Schema.org system has some limitations
  - XML-based schemas are too implementation specific
  - JSON Schema for now used in PlugFest to explore further
- Open issues
  - Semantic annotations alongside data structure definitions
  - Existing tool support for automatic validation

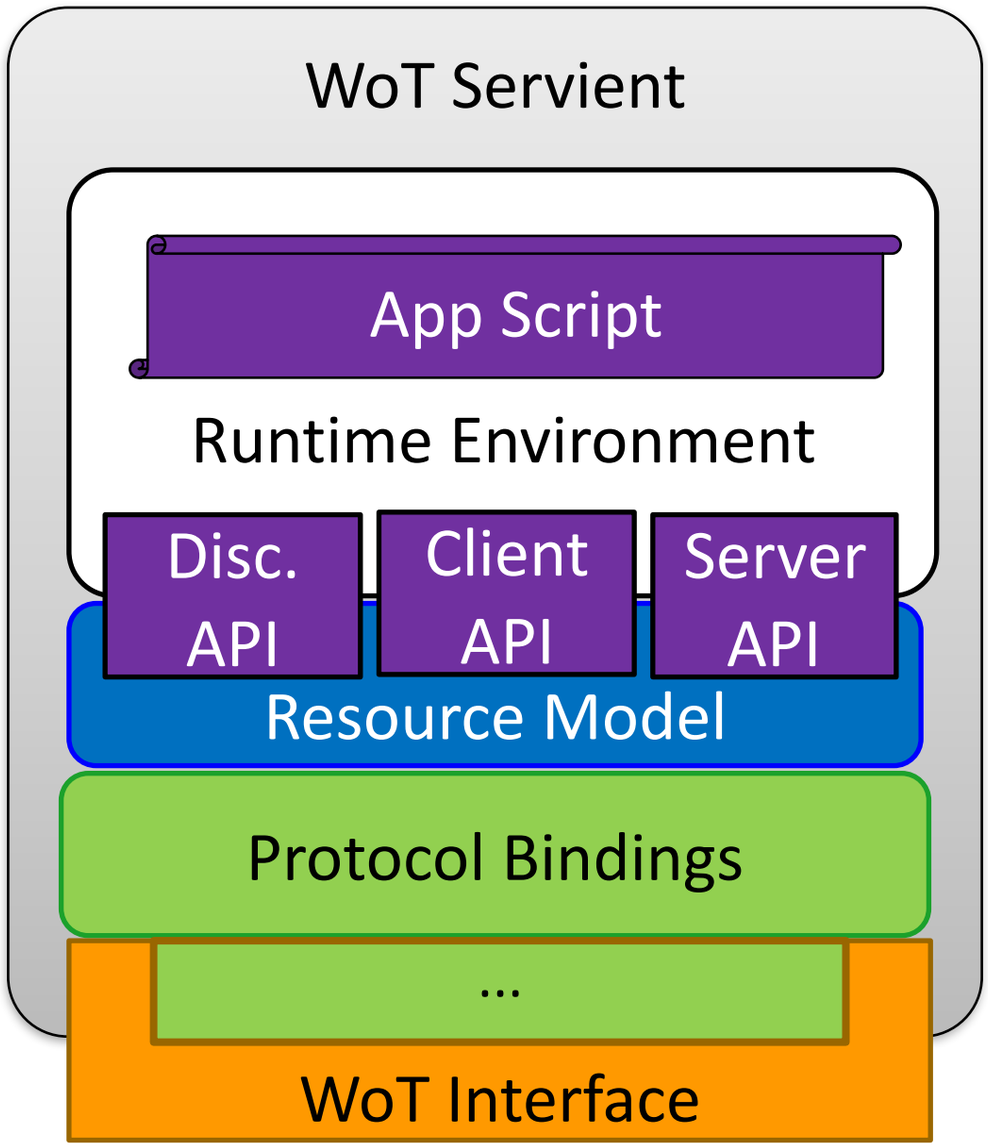
Without Scripting API

- Application logic often implemented natively



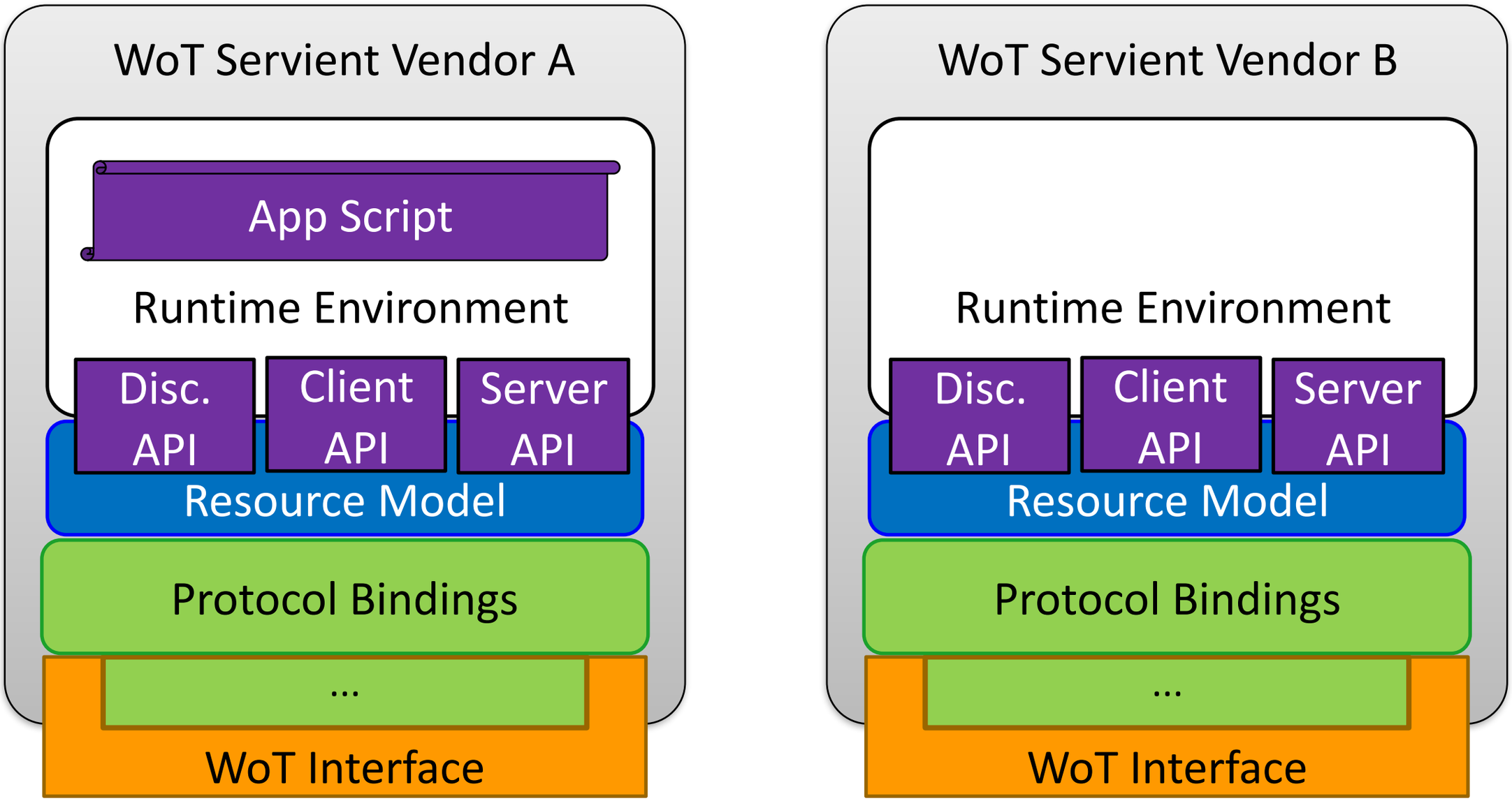
Scripting API

- Web-like development and deployment



Scripting API

- Common runtime enables portable apps



Script Example (Expose Thing)

```
// create software object to represent local Thing
WoT.newThing("counter")
    .then(function(thing) {
        thing
            // programmatically add interactions
            .addProperty("count", {"type": "integer"})
            .addAction("increment")
            .onInvokeAction("increment", function() {
                console.log("incrementing counter");
                // persistent state is managed by runtime environment
                var value = thing.getProperty("count") + 1;
                thing.setProperty("count", value);
                return value;
            })
        // initialize state (no builder pattern anymore)
        thing.setProperty("count", 0);
})
._catch(console.err);
```

Script Example (Consume Thing)

```
// create software object to represent remote Thing based on TD URI
WoT.consumeDescriptionUri("http://servient.example.com/things/counter")
    // use promise to handle asynchronous creation
    .then(function(counter) {
        counter
            // invoke an Action without arguments
            .invokeAction("increment", {})
                // which is an asynchronous call -> promise
            .then(function() {
                console.log("incremented");
                counter
                    // read Property (async.) to confirm increment
                    .getProperty("count").then(function(count) {
                        console.log("new count state is " + count);
                    });
            })._catch(console.error);
})
._catch(console.error);
```

W3C WoT F2F Beijing 2016

F2F MEETING AND PLUGFEST

F2F Meeting

- 11 – 14 July 2016
- Hosted by CETC in Beijing
  - Colocated with local IoT event
  - Exchange with CETC and local companies
- PlugFest and technical demos
- Plenary and breakout discussions

Scenario 1: Hello WoT

TD Web UI for human interaction

Servient platform with scripted apps

Servient connected to legacy devices

/voteTooHot

/on

Open Source

SIEMENS

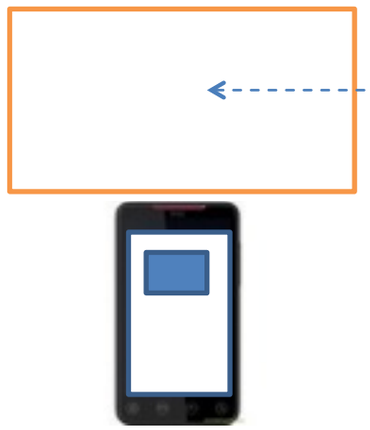
Panasonic

Scenario 2: Full WoT

Web Browser Scripting API

WoT Servient providing voter script and voting Servient

WoT Servient connected to legacy devices



/voteTooHot

/on

FUJITSU

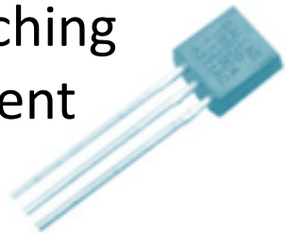
SIEMENS

Panasonic

/voteTooHot

TD Repository

WoT Servient searching for a voting Servient



Search for Action
@type="tooHot"

SIEMENS

Scenario 3: Rule-based Automation

Consume brightness sensor
to control curtain

SIEMENS

FUJITSU

PlugFest Online Resources

- Current Practices (Beijing Release)
  - http://w3c.github.io/wot/current-practices/wot-practices-beijing-2016.html
- Organization Wiki
  - https://www.w3.org/WoT/IG/wiki/F2F_meeting,_July_2016,_China,_Beijing#PlugFest
- Test Cases
  - https://github.com/w3c/wot/blob/master/plugfest/2016-beijing/plugfest-test-cases-beijing-2016.md
- Report Template
  - https://github.com/w3c/wot/blob/master/plugfest/2016-beijing/TestCaseCoverage.xlsx (t.b.d.)

Agenda

- 16:20 (Chairs) RG status update
- 16:30 (Chairs) Summary from RIOT Summit
- 16:45 Hannes, Stephen, Carsten:
  Summary from IOTSU IAB Workshop
- 17:15 Matthias Kovatsch:
  Update from W3C WoT IG and WG
- 17:35 (Authors) T2TRG documents
- 17:50 Tibor Pardi:
  Secure, decentralized, blockchain based IoT (talk)
- 18:10 (Chairs) Future activities

RESTful Design for Internet of Things Systems

draft-keranen-t2trg-rest-iot

Ari Keränen <ari.keranen@ericsson.com>
with Matthias Kovatsch & Klaus Hartke

T2TRG @ IETF96

Draft goals

- "Guidance for designing IoT systems that follow the principles of the REST architectural style"
- Collection of "basic" information and terminology that has been found useful

Next steps

- Application state
- Discovery mechanisms
- Resource design guidance
- Intro to hypermedia-driven apps

- But not much more. Publish.
  - Future docs on hyper-media aspects

Security consideration for the IoT

IETF96

Mohit (Ericsson)
Oliver (Siemens)
Sandeep, Oscar (Philips)

Contents in old draft-garcia-core-security-06

- Thing lifecycle
- Architectural considerations
- State of the art
- Challenges
  - Constraints
  - Bootstrapping
  - Operation
- Security profiles

Proposed way forward

- Thing lifecycle
- Architectural considerations <- Update
- State of the art <- Update
- Challenges
  - Constraints
  - Bootstrapping → refer to bootstrapping draft
  - Operation
  - New challenges (see next slides)
- (new) Solutions → bootstrapping solutions in bootstrapping draft
- Security profiles

Specific research topics to be added (1)

- Topics from: https://mailarchive.ietf.org/arch/msg/ace/Bgc3Mq3vxvOLi19fVR0ckbLOkuw
  - Firmware updates
  - Transparency and attestation of communications
  - Avoid device fingerprinting
  - Authorization handover (vendor)
  - Penetration testing

Specific research topics to be added (2)

- Further topics from https://github.com/t2trg/2015-ietf94/blob/master/t2trg-b.mkd
  - Handing over device ownership
  - *Lawful access*
  - Forensic readiness
  - Regulations and compliance
  - Cross-domain operation
  - …
- Others
  - Long term security

Proposed way forward

- Thing lifecycle
- Architectural considerations <- Update
- State of the art <- Update
- Challenges
  - Constrains
  - Bootstrapping → refer to bootstrapping draft
  - Operation
  - New challenges (see next slides)
- (new) Solutions → except bootstrapping solutions, those will be in bootstrapping draft
- Security profiles

- Sandeep
- Oscar
- Mohit
- Oliver

Q&A

Security considerations for the IoT

- starting from draft-garcia-core-security-06
- main contributors identified: Sandeep Kumar, Mohit Sethi, Jayaraghavendran K, Oliver Pfaff
- problems and guidelines (no completeness)
- cover lifecycle, ownership, stakeholders
- address recent IESG comments
- useful as a reference for security considerations sections in IETF standards

From https://github.com/t2trg/2015-ietf94/blob/master/99-t2trg-94-summary.pdf

A Survey of Security Bootstrapping Approaches

- starting from draft-he-6lo-analysis-iot-sbootstrapping-00
- main contributors identified: Mohit Sethi, Carsten Bormann, Yizhou Li, Robert Cragie
- application security vs. network security
- per-solution characteristics
- Grouping of solutions? Identifiable categories?

From https://github.com/t2trg/2015-ietf94/blob/master/99-t2trg-94-summary.pdf

Secure IoT Bootstrapping: A Survey

draft-sarikaya-t2trg-sbootstrapping-01

Behcet Sarikaya and Mohit Sethi

Secure Bootstrapping

- What is bootstrapping and what is secure bootstrapping? <- Updated
  - What is onboarding
  - What is identity and identifier
  - What is user and device identity and identifier

- Possible goals of secure bootstrapping:
  - Identity: authentication of a pre-established identity vs. creation of a new identity
  - Authorization for network access, incl. configuration of communication parameters
  - Registration or joining a domain or group
  - Pairing with a specific node, or connecting to a cloud service

- Some example of bootstrapping:
  - pairing of phones over bluetooth to exchange files, and
  - securely connecting IEEE 802.15.4 sensors factory to the backend both require some form of secure bootstrapping

Managed methods

- Pre-established trust relations and authentication credentials
- Centralized or federated
- Examples:
  - AAA / Extensible Authentication Protocol (EAP)
  - Generic Bootstrapping Architecture (GBA) with SIM
  - Open Mobile Alliance (OMA) Light-weight M2M:
    - Factory Bootstrap, Bootstrap from Smartcard, Client Initiated Bootstrap, Server Initiated Bootstrap
  - Kerberos
  - ANIMA <- Updated
  - Vendor certificates

P2P / ad-hoc methods

- No pre-established credentials
- Out-of-band channel used for distributing or confirming keys
  - Typically Diffie-Hellman exchange + MitM prevented with OOB communication
- Examples: <- Updated
  - Bluetooth simple pairing
  - Wi-Fi protected setup
  - EAP-NOOB (out-of-band authentication for EAP)
  - Magic wand, e.g. commissioning tool in I-D.kumar-6lo-selective-bootstrap

Opportunistic / leap-of-faith methods

- Continuity of identity or connection, rather than initial authentication
- Some methods assume that the attacker is not present at the inititial setup
- Examples: <- Updated
  - SEND and CGA
  - WPS push button
  - SSH, gmail, Facebook

Hybrid methods

- Most deployed methods are hybrid:
  - Components from both managed and ad-hoc methods
  - E.g. central management after ad-hoc registration
- Categorization is not always easy or clear
- Choice of bootstrapping method depends heavily on the business case:
  - What third parties available?
  - Who wants to retain control or avoid work?
  - Manufacturer/vendor, system admin, user, fully ad-hoc

Secure Bootstrapping

- Next steps:
  - Hidden gems and best practices?
  - Text on ownership transfer and how does it affect bootstrapping:

https://www.iab.org/wp-content/IAB-uploads/2016/03/draft-farrell-iotsi-00.txt

CoRAL and HSML

Media Types for Machine Interaction

Klaus Hartke and Michael Koster

Comparison

- Similarities
  - Collections of links and items
  - Forms to drive resource state updates
  - Interoperable data models
- Differences
  - CoRAL uses a data model derived from HAL
  - HSML uses CoRE Link-Format and SenML
  - CoRAL uses media types to define application semantic vocabulary and data serialization
  - HSML uses link annotation to embed application semantics

Next Steps

- Create a common use case prototype to evaluate both approaches
  - Cross-domain interoperability
  - How does the difference in semantic annotation impact application design?
  - Discovery, resource construction, application interaction
- Converge to a single representation format and interaction model over time

The BLE (Bluetooth Low Energy) URI Scheme and Media Types

draft-bormann-t2trg-ble-uri-00

Carsten Bormann & Ari Keränen

T2TRG @ IETF96

Background

- Bluetooth Low-Energy (BLE): popular technology for constrained devices
- Resources of BLE devices can be accessed over IP (RFC7668) or via gateways
- How about locally connected devices and web technologies?
- Straw man proposal of BLE URI scheme and media types

Example

- Passive scan for nodes:

```
GET ble:/gap/nodes/passive
```

- ..results in node list; used for query services

```
GET {node}/services
```

- ..returning "application/ble-gatt-servicelist"

```
servicelist = [* service]
service = {
  href: text,
  uuid: uuid,
}
uuid = bytes .size 16
```

Next steps

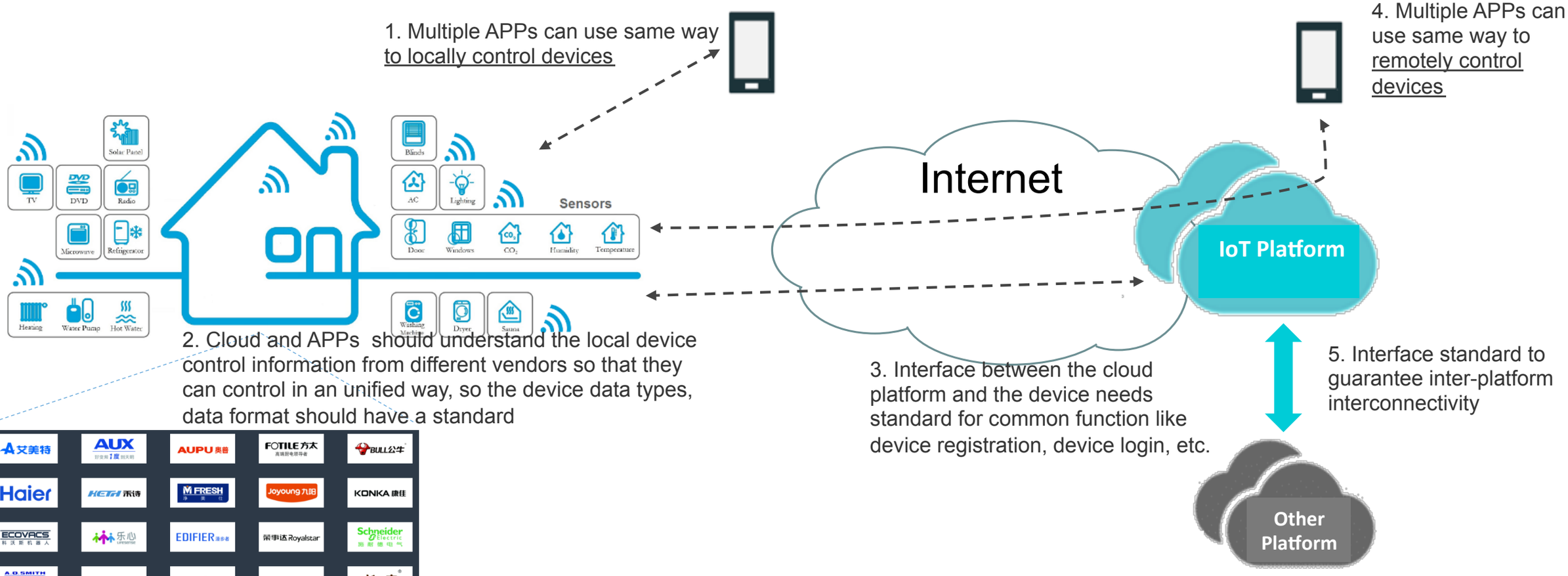
- Adding (much) details
- Align with Web Bluetooth
- Reviews from Bluetooth experts

IoT Platform Architecture and Data Model

https://www.ietf.org/id/draft-liu-t2trg-architecture-data-model-00.txt

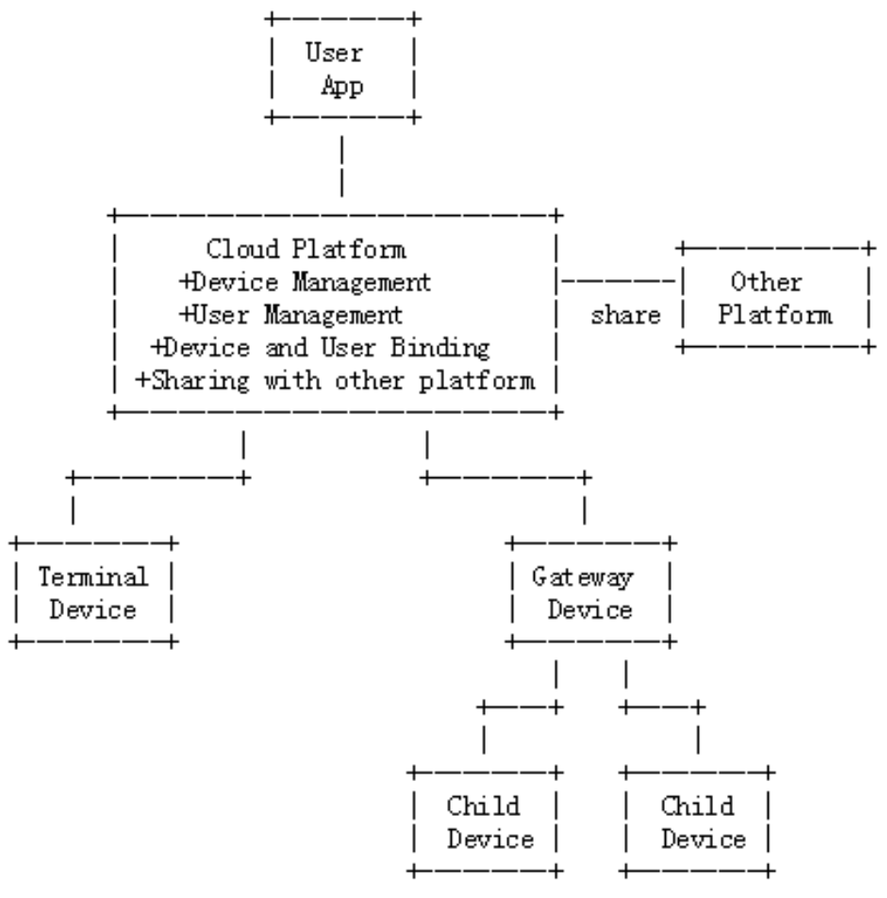
Dapeng Liu
Alibaba Group

The Smart Home Ecosystem



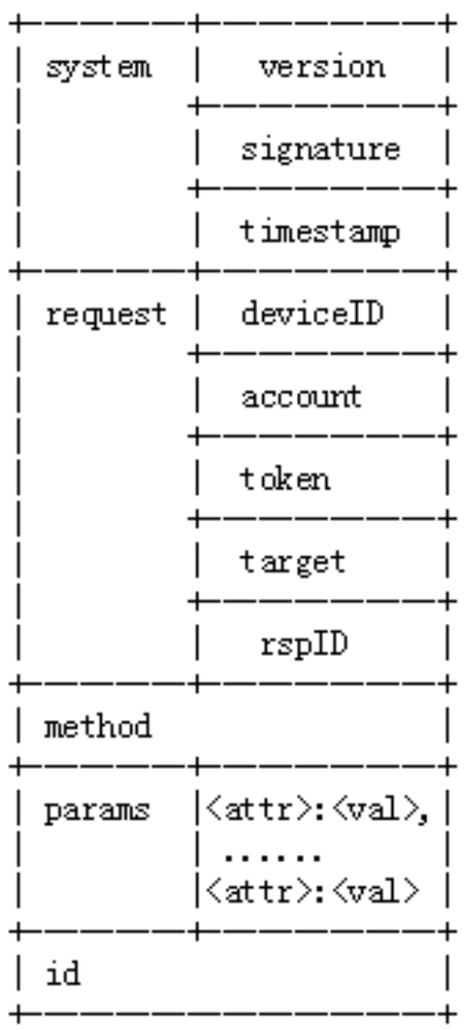
Connecting with multiple device vendors

Data Model Design for IoT Platform



- The data model can be applied to various kinds of IoT service platform scenarios, example：smart home

Data Model



- Can be used in the communication between service platform and user APP, between service platform and other platform, between service platform and IoT devices, and between service platform and gateway device
- Default encoding schema for this data model is JSON

Fields in Data Model

Name	Format	Length	Description
version	String	0-255	Data model version
signature	String	32-255	Signature value
timestamp	String	0-255	Timestamp
deviceID	String	0-255	Optional, required when data is sent by device
account	String	0-255	Optional, required when data is sent by user application, or server, or other vendor's platform
token	String	0-255	Optional, required when data is sent to server. The token is assigned by server to device, user, or vendor platform
target	String	0-255	Optional, required when data is sent to server, indicating target destination
rspID	String	0-255	Optional, required when data is a response to last remote control command data. The value is set to last command data's id filed value
method	String	0-255	Indicate the method
params	String	0-1023	Attribute set
id	String	0-255	message ID

Examples


```
{
        "system":{
                "version":"1.0",
                "signature":"5eeff300d71f13610f283d36b4f16ffa",
                "timestamp":"1407543671"
        },
        "request":{
                "deviceID":"35595459BDD240E029C40033C4B69F16",
                "token":"zzzxxxyyzzmmmssssiiiioopppqqq",
                "target":"user00000001",
                "rspID":"100"
        },
        "method":"postDeviceData",
        "params":{
                "temperature":{
                        "value":"34.8",
                        "when":"1404443289"
                },
                "humidity":{
                        "value":"45",
                        "when":"1404443359"
                }
        },
        "id":"91"
}
```


One example that device posts data to server

Examples


```
{
        "system":{
                "version":"1.0",
                "signature":"3grff300d71f13610f283d36b4f16ffa",
                "timestamp":"1404443389"
        },
        "request":{
                "token":"xy87799923eerueirueio",
                "target":"29C40033C4B69F1635595459BDD240E0"
        },
        "method":"getDeviceStatus",
        "params":{
                "uuid": "35595459BDD240E029C40033C4B69F16",
                "attrSet":[
                        "temp",
                        "humidity"
                ]
        },
        "id":"100"
}
```


One example that user APP requests server to get device status

Agenda

- 16:20 (Chairs) RG status update
- 16:30 (Chairs) Summary from RIOT Summit
- 16:45 Hannes, Stephen, Carsten:
  Summary from IOTSU IAB Workshop
- 17:15 Matthias Kovatsch:
  Update from W3C WoT IG and WG
- 17:35 (Authors) T2TRG documents
- 17:50 Tibor Pardi:
  Secure, decentralized, blockchain based IoT (talk)
- 18:10 (Chairs) Future activities

Decentralized, peer-to-peer IoT

MANAGE IOT DEVICES WITH BLOCKCHAIN BASED, PEER-TO-PEER, DECENTRALIZED SYSTEMS

Who we are?

- Group of open source developers
- We do blockchain and decentralized, P2P application development
- We develop Streembit http://streembit.github.io/
- We participate in the W3C standardization process

The Problem

Problems with proprietary, closed source client-server systems

- Security and Privacy, mitigate the risk of inside job hacking
- Economy
- Politics - Incoming communication legislation such as the UK Investigatory Powers Bill

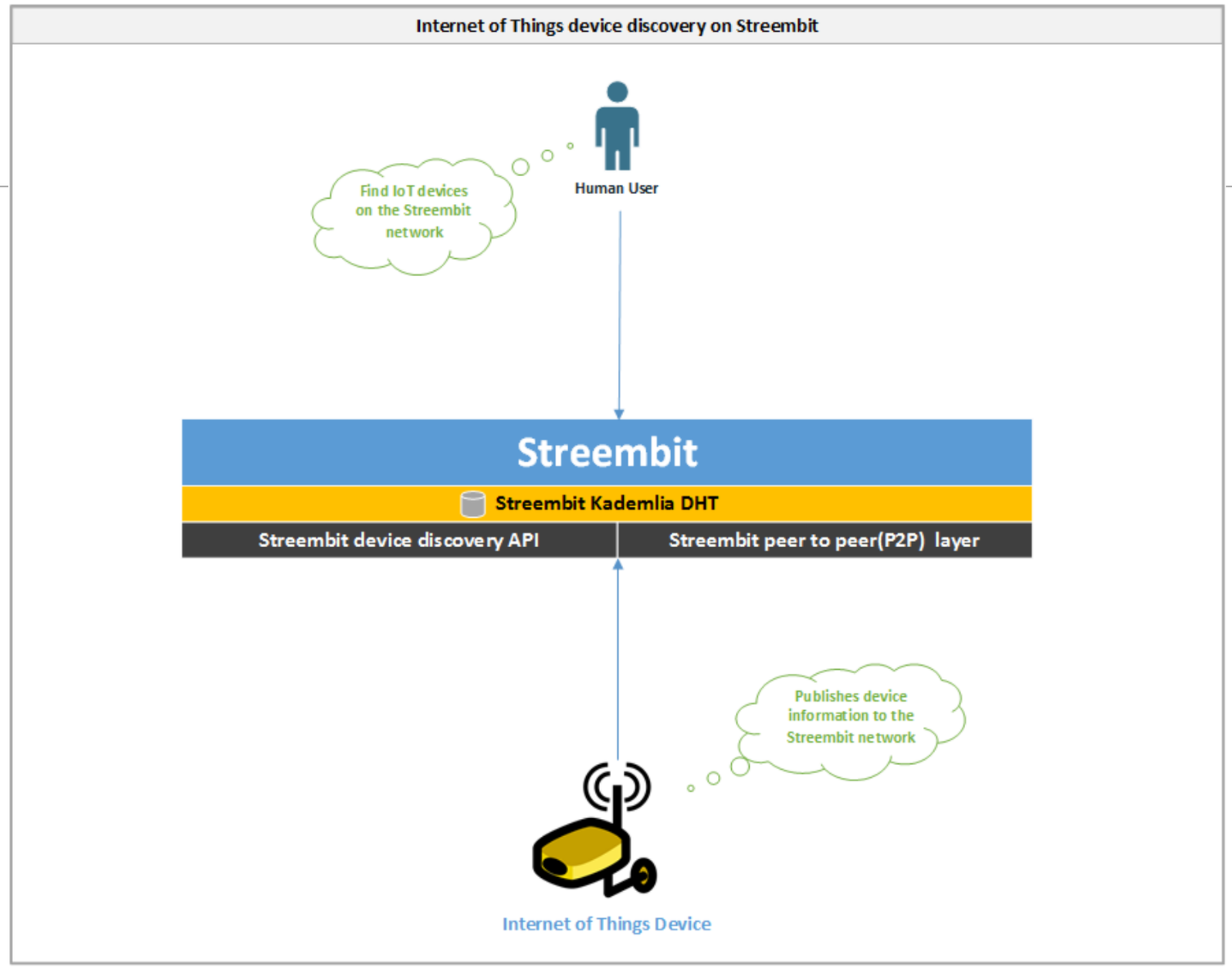
The solution

Use decentralized, peer-to-peer systems to move away from the cloud.

Blockchain technologies:

- Confirming data origin and accuracy
- Tracking updates
- Establishing true data authority for millions of different data fields
- Smart contract management

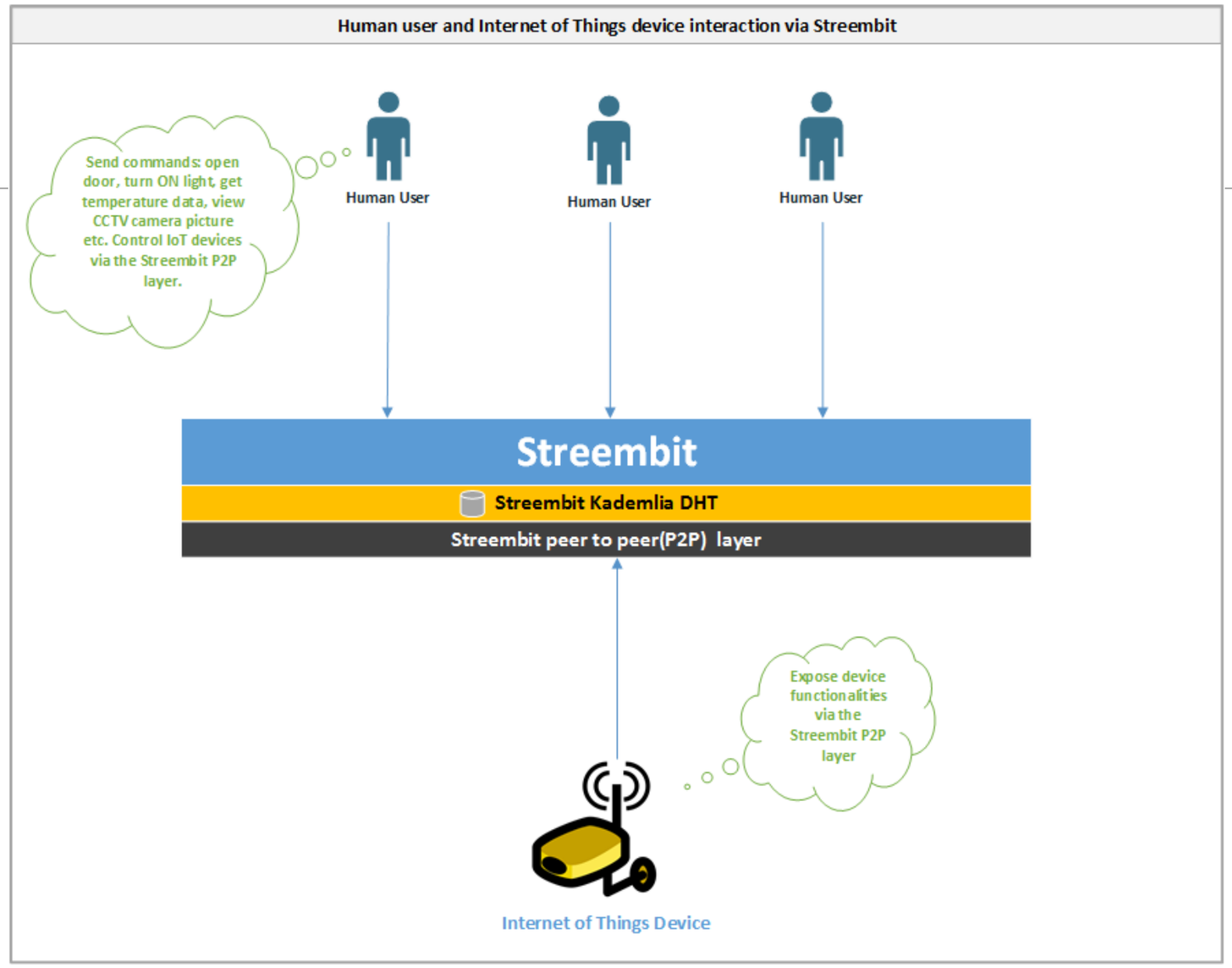
Device Discovery



Control Internet of Things devices

- Via peer to peer manner
- End to end encrypted between the human users and IoT devices
- Using W3C WoT standards

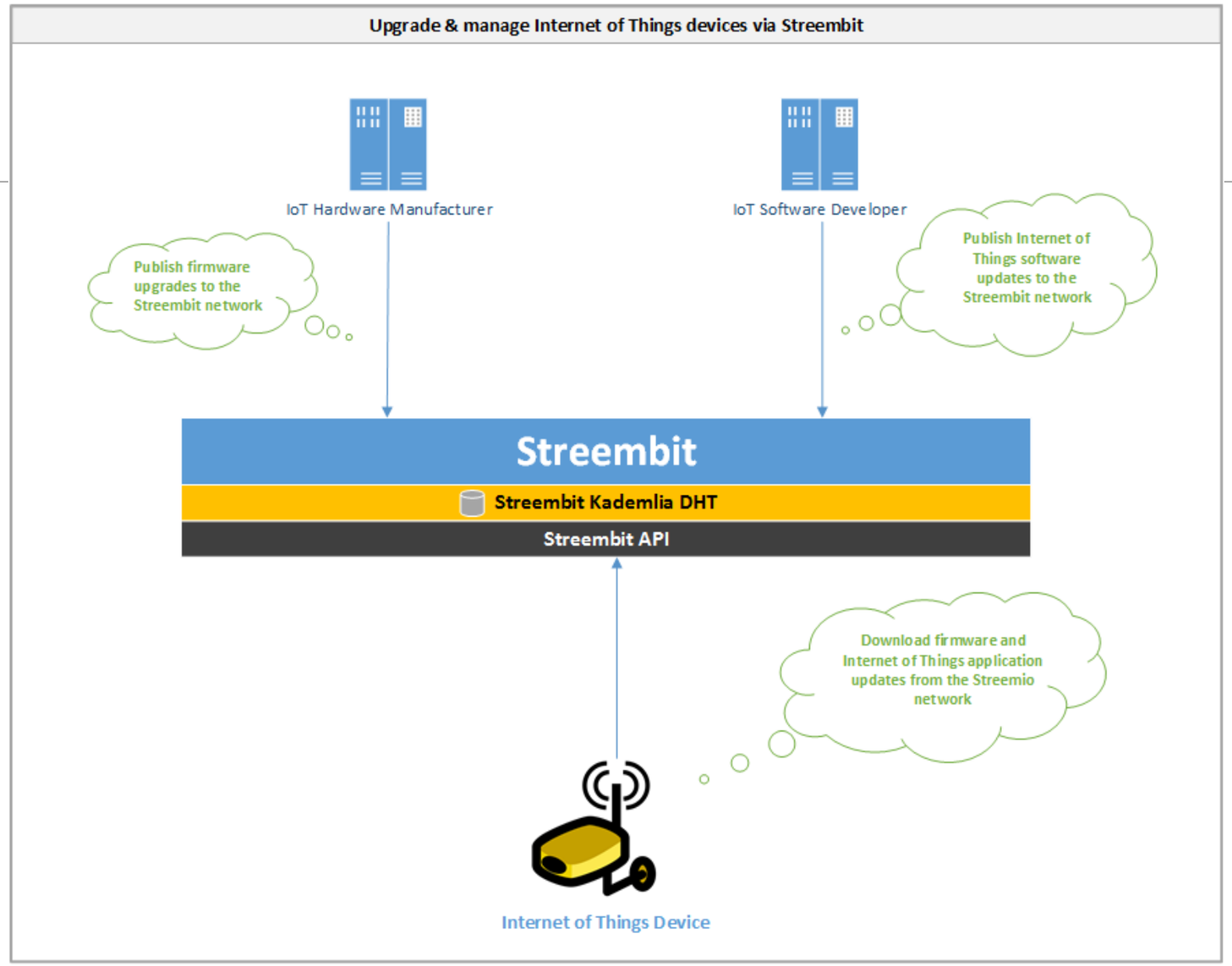
Control Internet of Things devices



Upgrade and manage IoT devices

- Hardware and software providers upgrade Internet of Things devices on the always up and running on decentralized networks.
- Internet of Things device manufacturers and software designers publish firmware and software updates via the decentralized network.
- Ensure via strong PPKI security that the origin and data integrity of the updates by verifying the public key of the publisher.

Upgrade and manage IoT devices



Strong security

- Based on PPKI, ECC cryptography
- Each actor of the system must generate a public/private key pair. (Typically keys are generated prior to configuring the device and will be burned into the devices' firmware).
- The devices and users publishes the public key to other users of the system.
- The data integrity and authenticity of the messages is guaranteed with PPK signatures.
- Each session between users is secured with strong 256-bit AES symmetric symmetric cryptography keys.
- Uses ECC Diffie Hellman (ECDH) key exchange

Working on standards

We try to create an IETF standard for decentralized, peer-to-peer IoT.

Github protocol repository

Contact info

Tibor Zsolt Pardi

tzpardi@streembit.com

http://streembit.github.io/

Skype: zsolt.pardi

Agenda

- 16:20 (Chairs) RG status update
- 16:30 (Chairs) Summary from RIOT Summit
- 16:45 Hannes, Stephen, Carsten:
  Summary from IOTSU IAB Workshop
- 17:15 Matthias Kovatsch:
  Update from W3C WoT IG and WG
- 17:35 (Authors) T2TRG documents
- 17:50 Tibor Pardi:
  Secure, decentralized, blockchain based IoT (talk)
- 18:10 (Chairs) Future activities

Next meetings

- **SDOs: Co-locate with W3C WoT meeting @ TPAC in Lisbon (Thu/Fri Sep 22/23): Sat/Sun Sep 24/25**
- Open-Source: October Eclipse?
- Full meeting in Seoul before IETF97 (Sat/Sun Nov 12/13)?
- Academic: February @EWSN?