Named Data Networking of Things: NDN-IoT Framework & "FLOW"

Zhehao Wang, Eitan Mendelowitz,
Zoe Sandoval, Jeff Burke

Jan 30, 2017

Portions from Alex Afanasyev's presentation at IoTDI 2016

Framework code includes components by Jeff Thompson

Outline

1. Background on NDN team's approach to IoT
2. NDN-IoT, an IoT framework over NDN, in JavaScript, Python, C# and C++, based on the NDN Common Client Libraries. (LGPL v3)
3. "Flow", a prototype home entertainment application for multiple users built with NDN-IoT. (GPL v3)… Alpha stage!

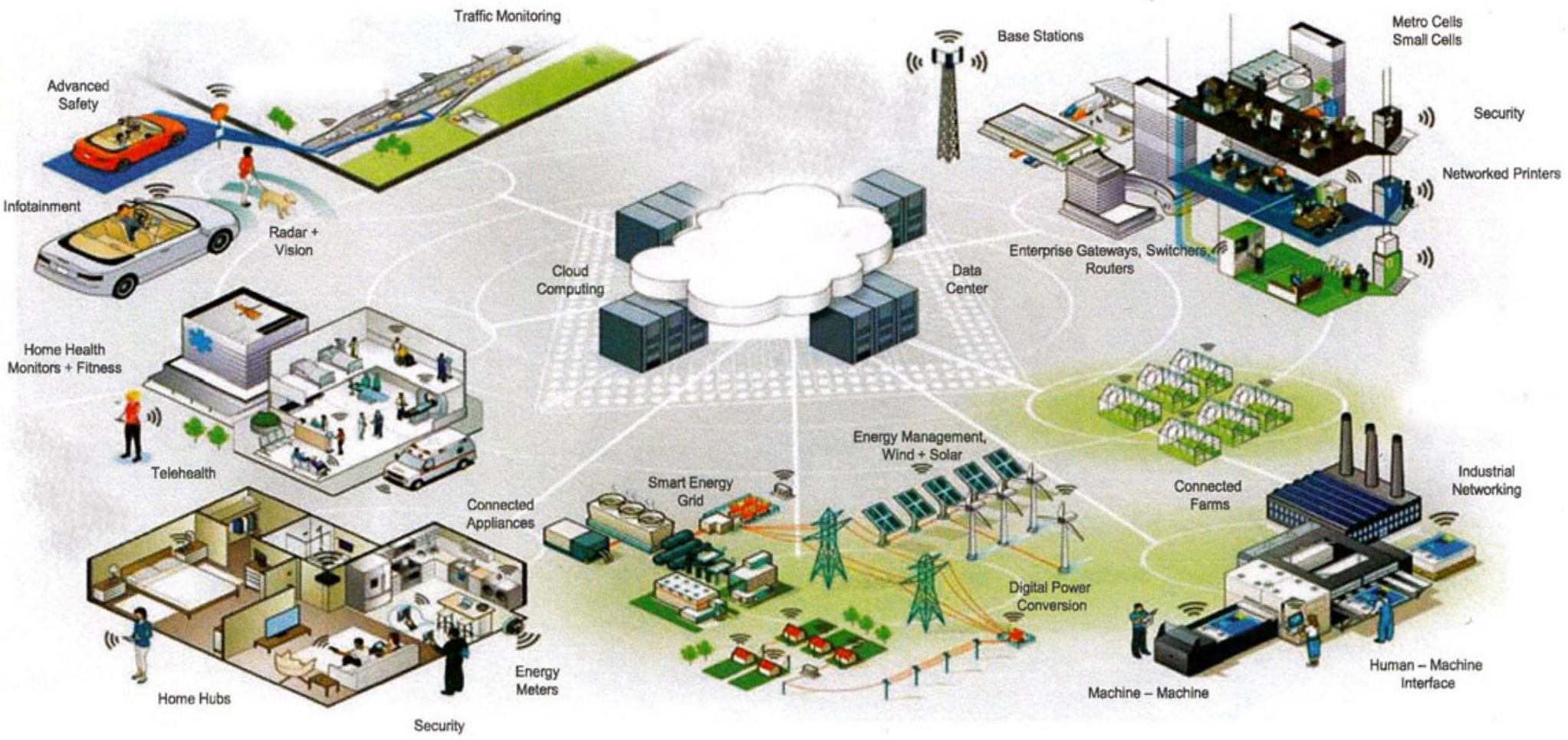
Flow: introduction

- Flow is a prototype home entertainment application built on top of NDN-IoT framework. Players
  - Interact with the game by walking around in an area tracked by OpenPTrack (opt)
  - Sees physical tracks affect the terrain in a virtual space rendered by Unity3D game engine
  - Drop social media images to the virtual space using a mobile webpage
  - Collectively control the angles the virtual cameras are facing by rotating gyroscopes

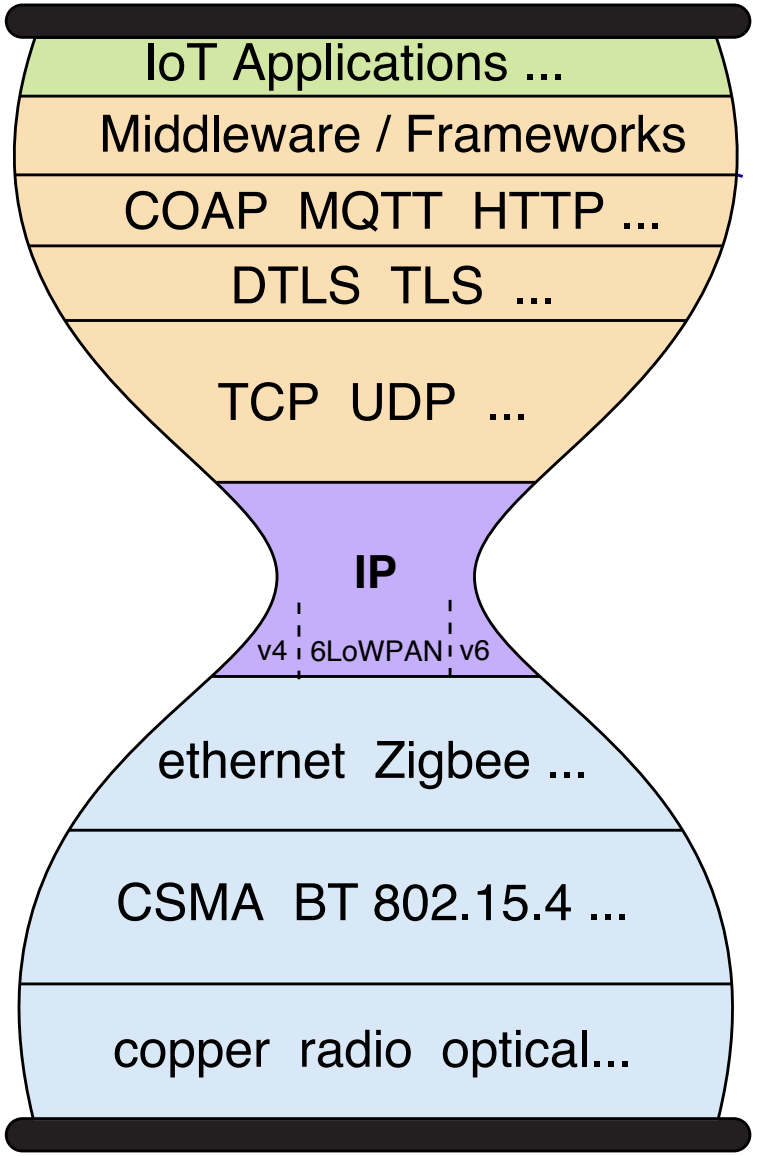
Flow: motivation

- Simple driver app for ideas in IoTDI 2016 paper
- Explore and demonstrate integration
  - Game engine (3D graphics)
  - Infrastructure-based sensing (home game kit)
  - Wearable IoT devices (Arduino class)
  - Mobile phones
- Explore ideas for an interactive public project
- Multi-platform, multi-language, multi-user

Vision of the Internet of Things (IoT)



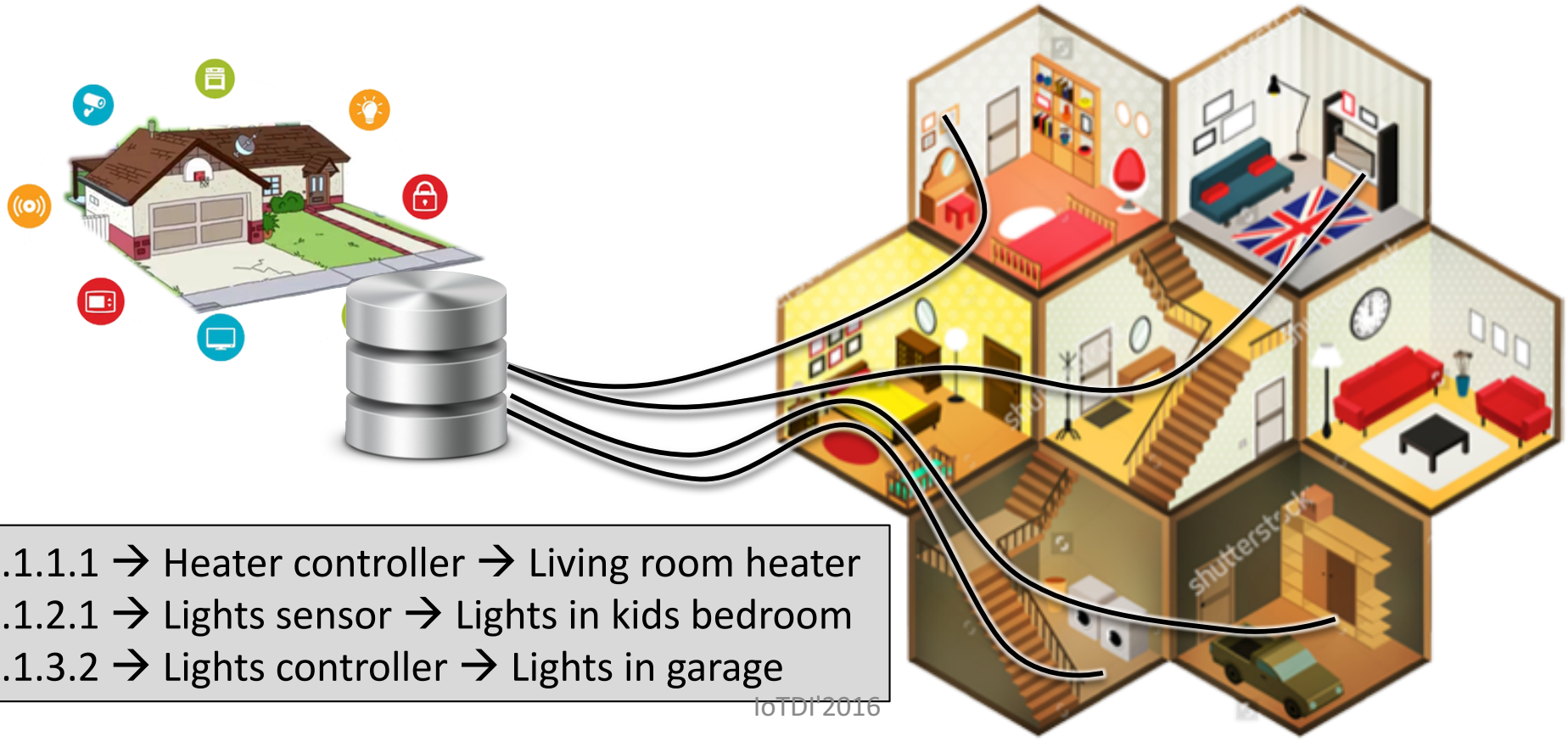
Today's IoT over TCP/IP



- Lots of engineering effort to patch TCP/IP for IoT
  - DNS for name-IP mapping
  - App-layer bridging of semantic gap
- Channel/session-based security or physical isolation
- Multiple network technologies and need for gateways
  - 6LoWPAN for 802.15.4 networks
  - IPv6-over-foo adaptation layers for different L2 technologies
- Poor integration of local communication, ability to use intermittent links

Today's IoT over TCP/IP

- Point-to-point communication model
- With focus on devices that are associated with a "things", not "things" themselves



1.1.1.1 → Heater controller → Living room heater
1.1.2.1 → Lights sensor → Lights in kids bedroom
1.1.3.2 → Lights controller → Lights in garage

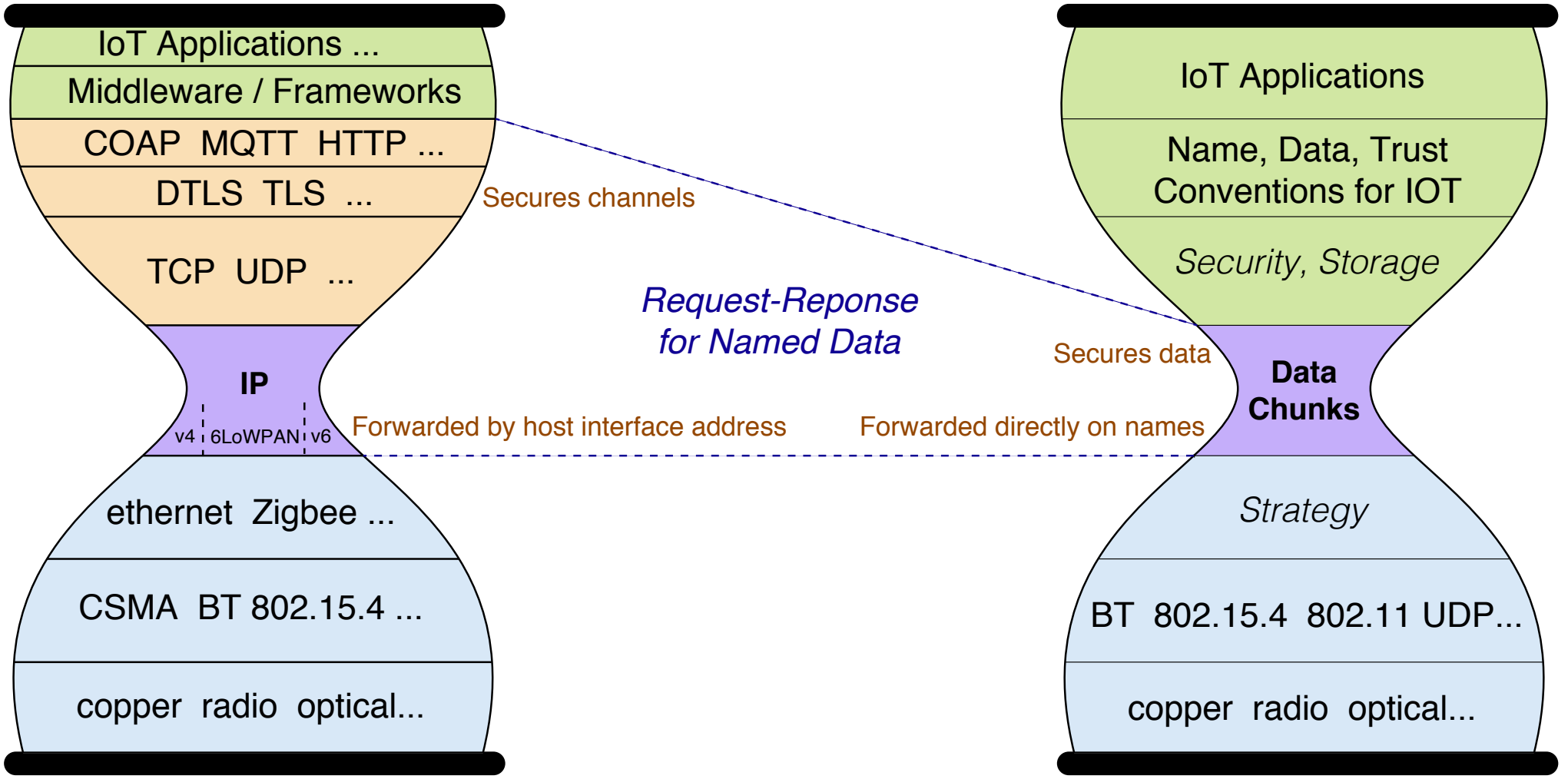
Reasons for Today's Challenges

- TCP/IP designed to interconnect mainframes
  - We have done great engineering to patch up the protocol stack to do lots things it's not designed for, bridging the gap between what TCP/IP offers and what apps need.
- More layers added to get closer to the IoT semantics
  - More complexity and solution diversity
- IoT is fundamentally different from all the other apps before it
  - many devices
  - constrained processing power, memory size, battery life
  - usually constrained and intermittent connectivity
  - too critical to go without security (or cloud based security)

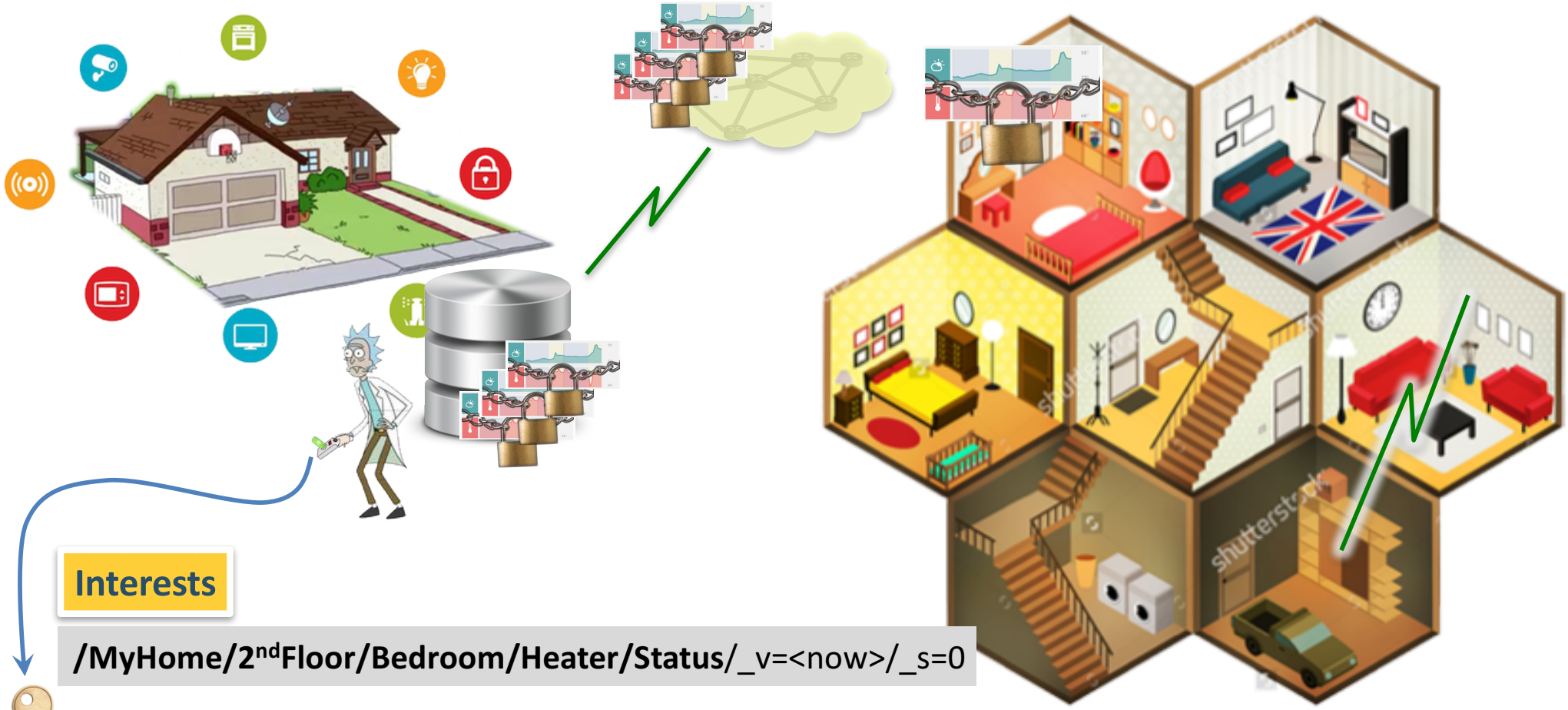
What We Need to Enable IoT Vision

- Universal network protocol to interconnect all networking technologies
  - Ethernet, WiFi, 802.15.4, Bluetooth, Zigbee, USB, Serial, ...
- Move IoT semantics to the network layer
  - Name the “things” and operations on “things”
    - “temperature in the room”, “humidity on the second floor”
    - “blood pressure”, “body temperature”
    - “max/min/avg pH of soil in specific point of US soil grid”
  - Focus on data associated with things, not devices
    - status information or actuation commands
  - Secure data directly

Named Data Networking of Things



NDN Smart Home



Interests

/MyHome/2nd Floor/Bedroom/Heater/Status/_v=<now>/_s=0

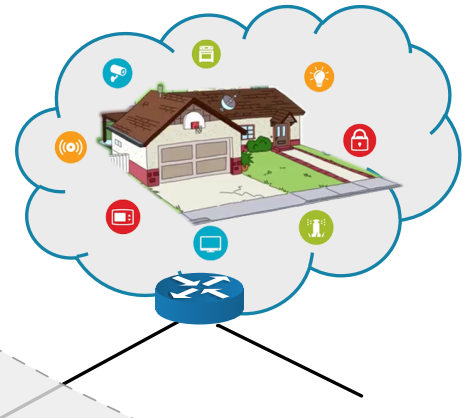
Interests

/MyHome/Garage/Lights/ON/<signature>/...

Accessing NDN Smart Home

Interests

/MyHome/…

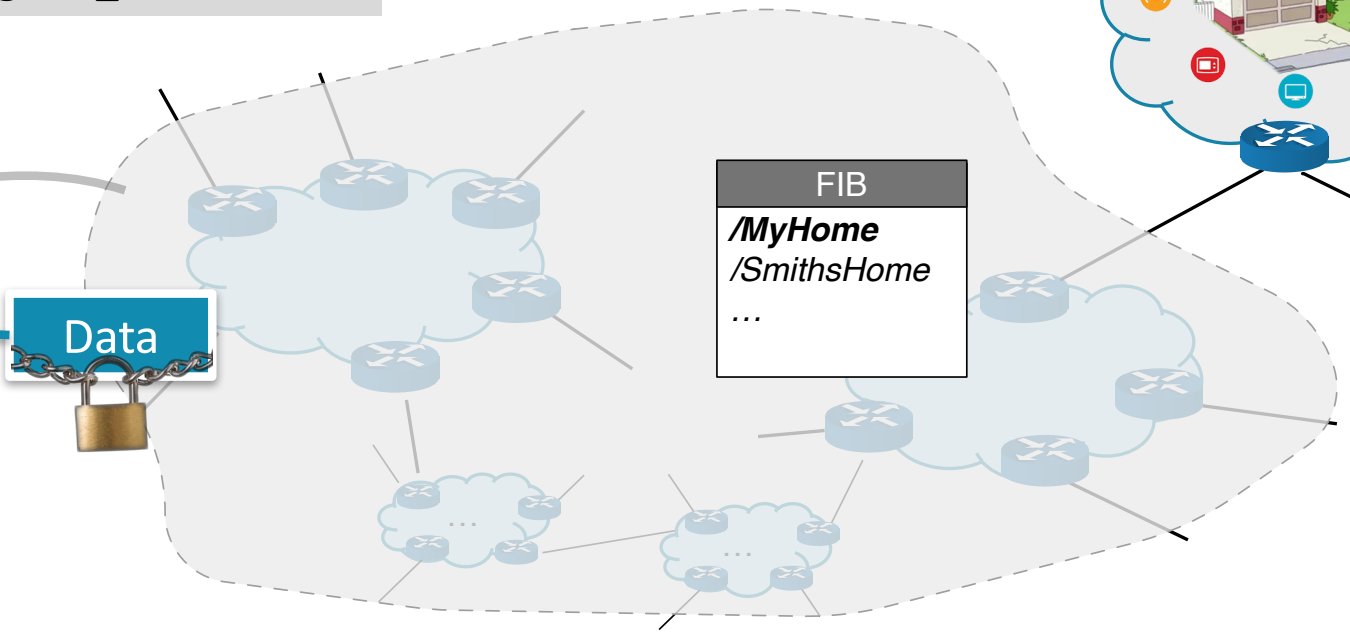


/MyHome/2nd Floor/Bathroom/Heater/Status/…

/MyHome/Basement/Lights/_cmd=OFF/…

Interests

Data



FIB

/MyHome

/SmithsHome

…

Accessing NDN Smart Home
NDN Overlays

Interests

/MyHome/…

/MyHome/2nd Floor/Bathroom/Heater/Status/…

/MyHome/Basement/Lights/_cmd=OFF/…

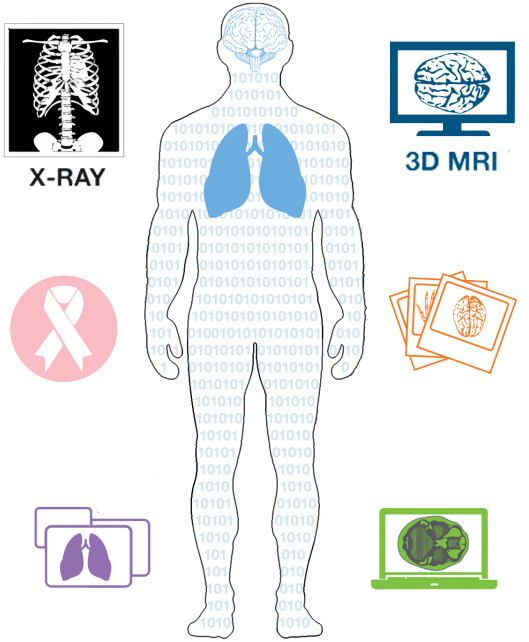
Interests

Data

NDN Personal Health

/MyHealth/me/FITNESS/steps/20160301/_v=121/...

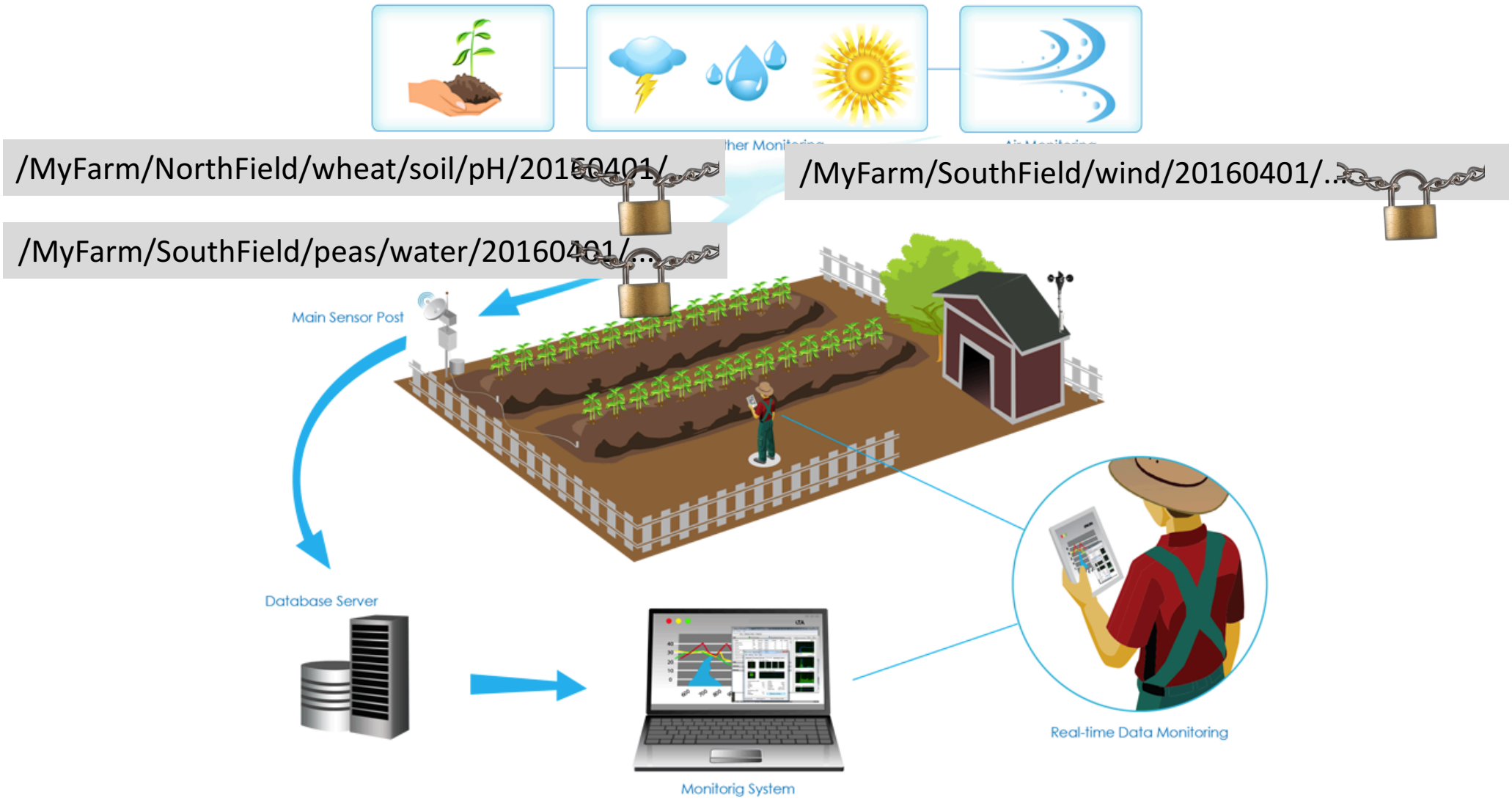
/MyHealth/me/FITNESS/steps/20160301/_v=121/...



/MyHealth/me/LABS/xray/20160301/_v=121/...

/MyHealth/me/SHOTS/flu/20151215/_v=42/...

NDN Precision Agriculture



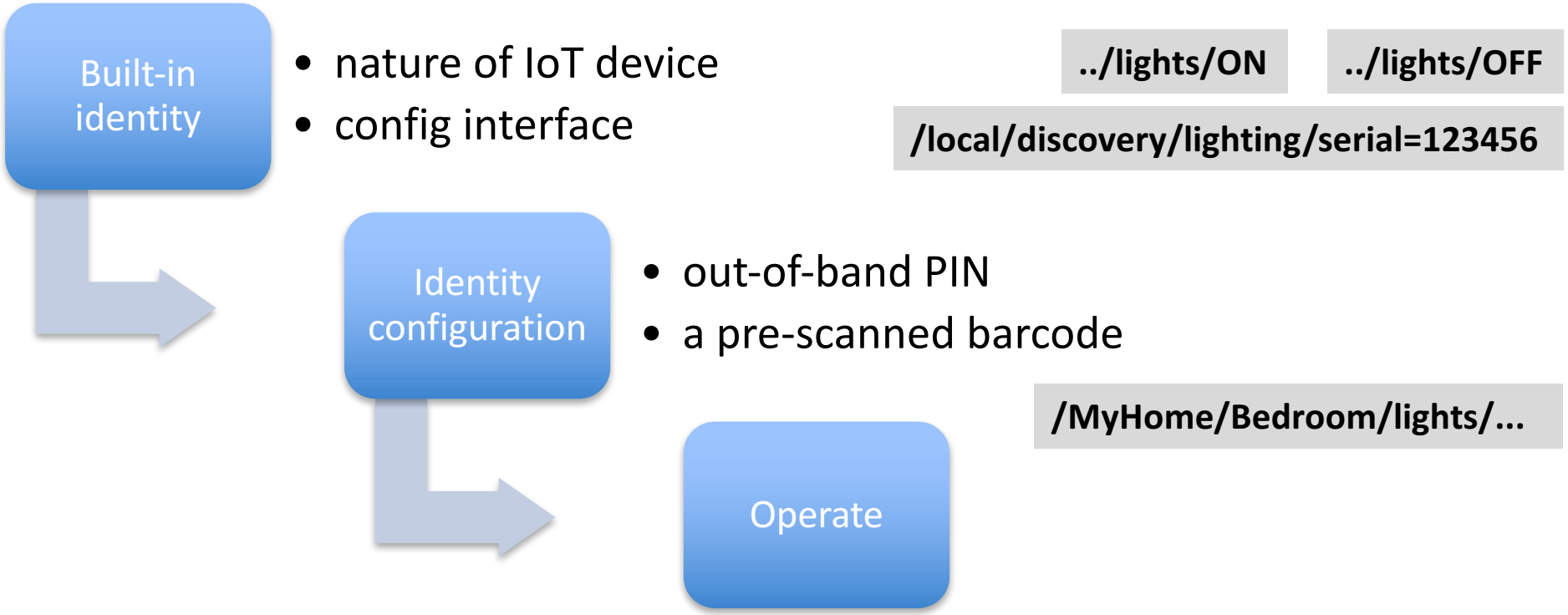
NDN Alignment with IoT Applications

- Request-response semantics
- Data-centric security
- Name-based forwarding
- In-network cache

- Application semantics at the network layer
- Make use of ad hoc and broadcast-style communications
- Make use of any intermittent connectivity
- Independence of communication technology

Bootstrapping, discovery, and auto-config

- Bootstrapping, discovery, and auto-config
  - no IP address allocations / ma needed



Other Elements of NDN IoT Framework

- Schematized trust and automated verification
- Name-based access control
- Pub-sub communication mechanism
- Data synchronization
- Integration with global Internet

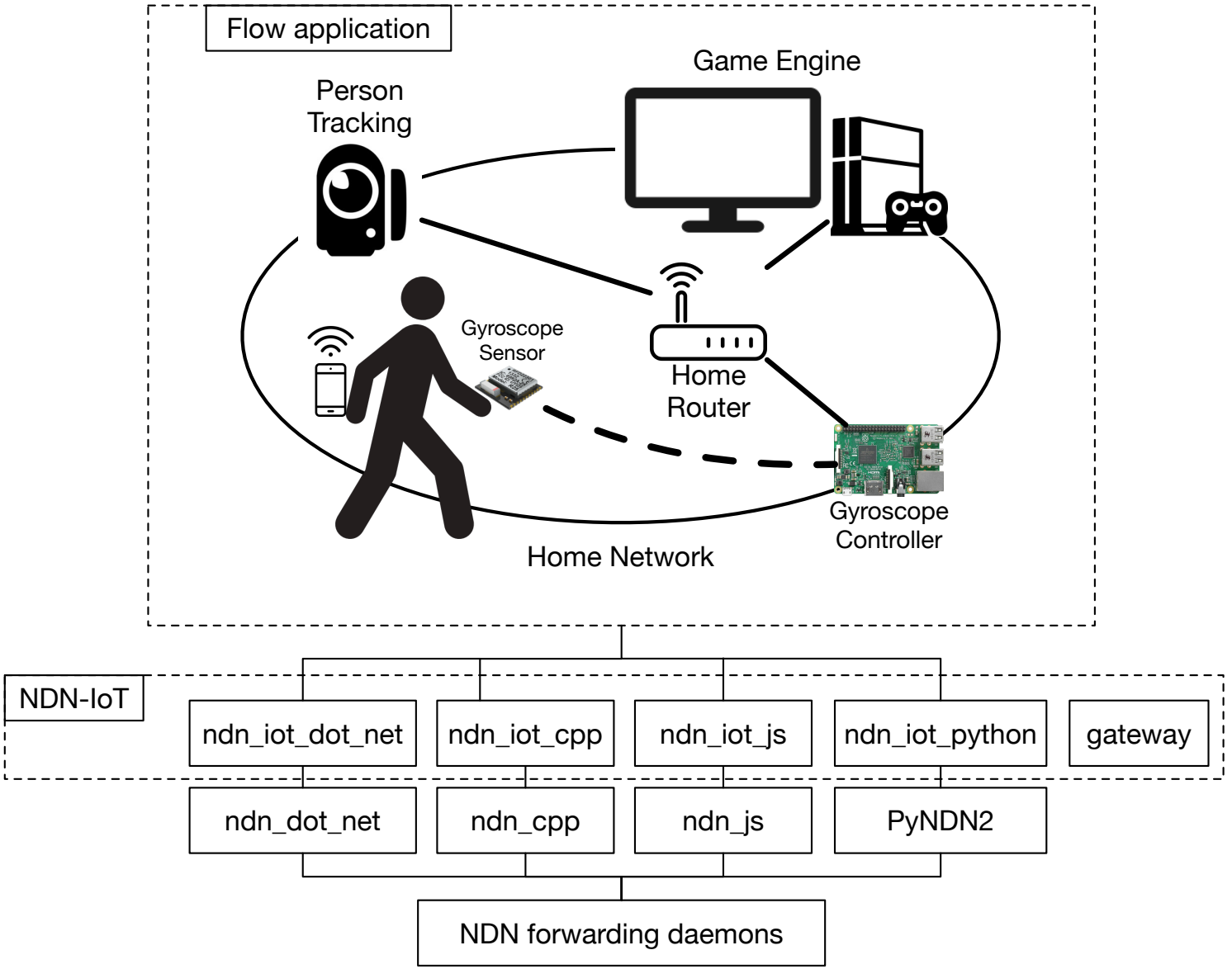
Challenges

- Naming design
  - Network and security considerations at app design level
  - Potential need for multiple naming hierarchies
- Constrained devices
  - Which subset of NDN features must constrained devices support
  - Considerations include processing power, battery power, and memory capacity
- Constrained networking
  - Considerations to low-rate low-MTU packets
- Routing & forwarding in ad-hoc environment

FLOW / NDN-IoT

- First pass on a multi-platform application
- Focus on IP-IoT challenges
  - Complex solutions to simple communication needs
  - Limitations of channel- and session-based security
  - Poor integration of local communication
- Progress on proposed solution
  - Shang et al. **Named Data Networking of Things.** *IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI),* April 2016.
  - Shang et al. **A Cloud-independent Home Entertainment Design over Named Data Networking of Things**. *ACM/IEEE Second International Conference on Internet-of-Things Design and Implementation (IoTDI)*, April 2017.

Introduction – NDN-IoT & Flow



NDN-IoT Framework

NDN-IoT: functionality

- NDN-IoT (Named Data Networking of Things) is a set of libraries implementing functionality described in the NDN team's IoTDI 2016 paper [1]
  - Naming things, devices, and their data (VI.A, VI.H)
  - Schematized trust (VI.C)
  - Device and application bootstrap (VI.B, VI.C)
  - Device/service/capability discovery (VI.B)
  - Application-level pub/sub (VI.F)

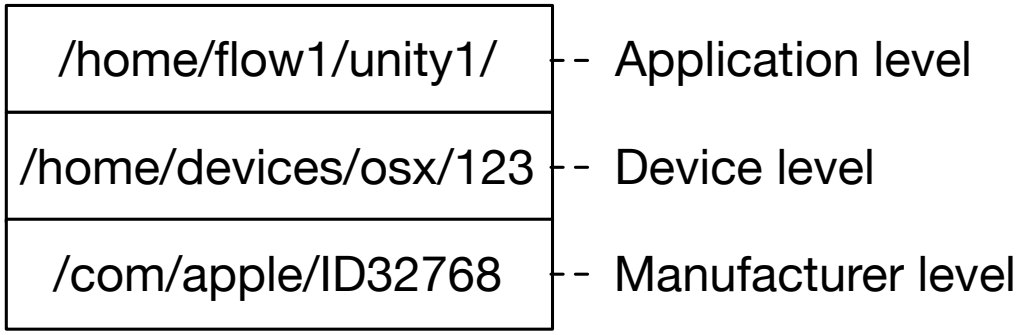
[1] W. Shang, A. Bannis, T. Liang, Z. Wang, Y. Yu, A. Afanasyev, J. Thompson, J. Burke, B. Zhang, L. Zhang. Named Data Networking of Things (Invited Paper). In IoTDI, 2016

NDN-IoT: naming

- The framework can support arbitrary names. In our application we used three levels of names
  - Application/thing level: name the thing
    - An application-specific “label” given by the user
    - For application data exchange
  - Device level: name the device in a space
    - Some physical properties of the device to identify the device
    - For device control and status feedback
  - Manufacturer level: name given by the manufacturer
    - For initial device profile retrieval and verification

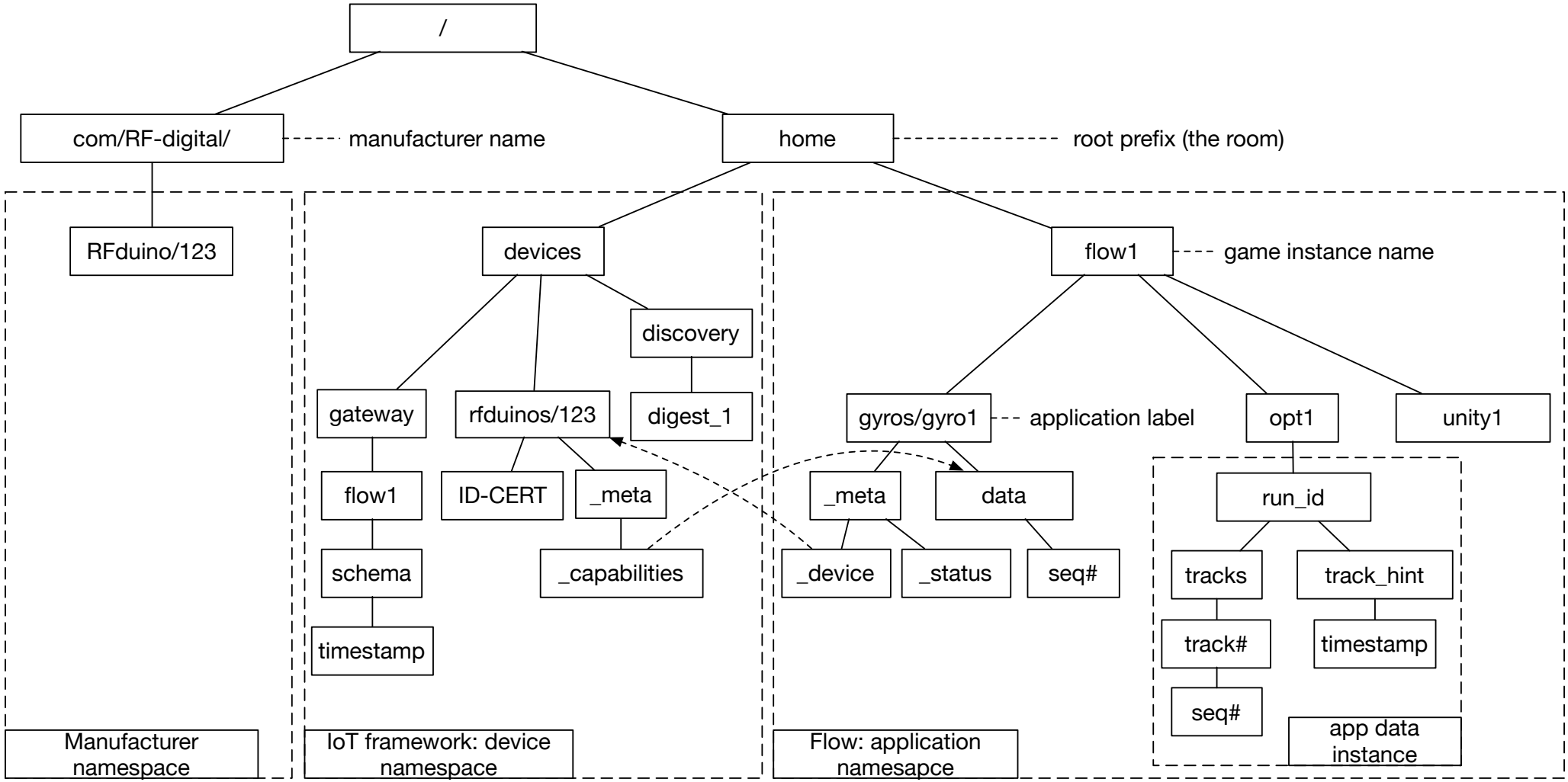
NDN-IoT: naming – cont

- An example of the three levels of names



- The framework suggests this naming convention to
  - Decouple data from the device publishing it
  - Reflect the trust relationship in a home IoT environment

Namespace – A work in progress



NDN-IoT: global Internet integration

- Global Internet integration can be achieved using
  - A globally reachable prefix such as "/ucla/MelnitzHall/1469C", or
  - A local prefix such as "/home", with the help of forwarding hints* or encapsulation**

** Forwarding hints in NDN FAQ*

*** Map-and-encap discussed in NDN-TR004 rev2*

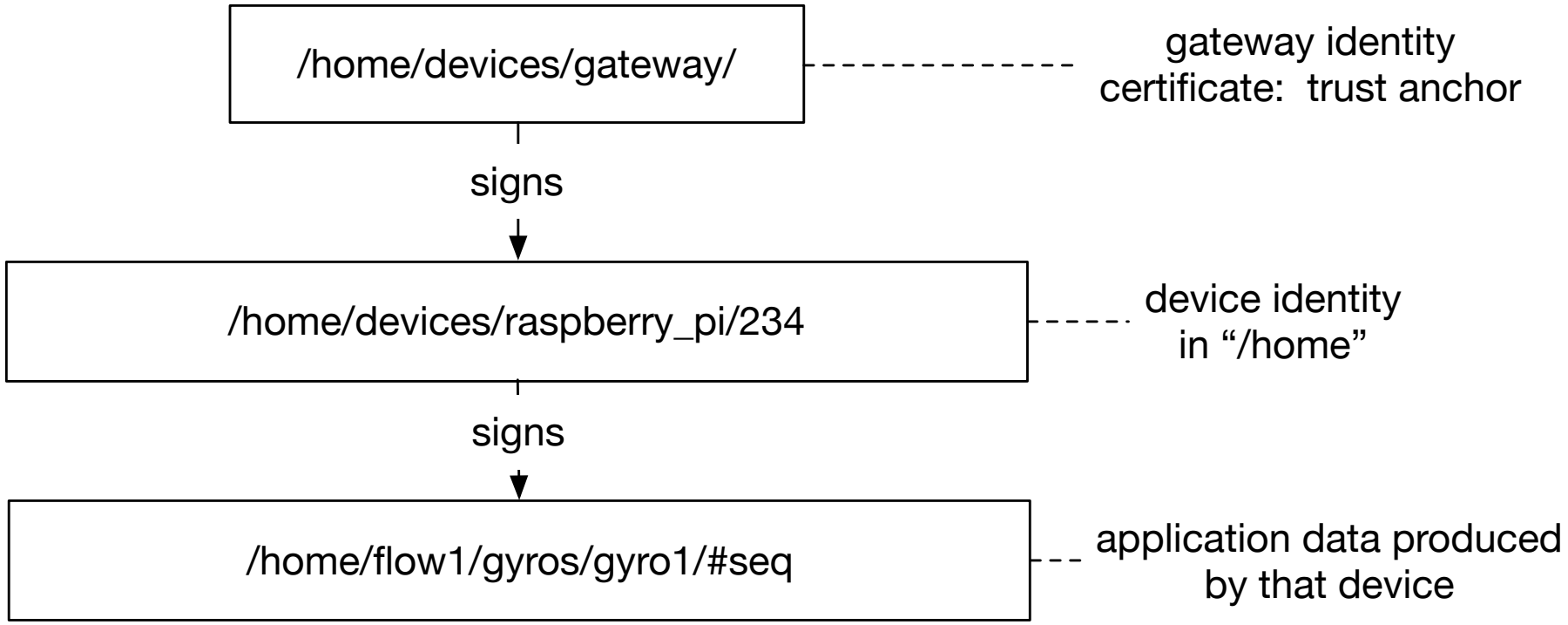
NDN-IoT: trust

- NDN-IoT uses schematized trust [3], which
  - Leverages the structure of names to schematize decision-making on a packet-by-packet basis
  - Uses a schema file to define expected relationship between a data name and a KeyLocator key name, and forms a hierarchical structure
- The framework uses a hierarchical trust model
  - A gateway serves as the trust anchor
  - The gateway's key signs device certificates (device namespace) (RSA)
  - Each device's key signs application data it produces (application namespace) (RSA, HMAC (constrained devices))

[3] Y. Yu, A. Afanasyev, D. Clark, K. Claffy, V. Jacobson, and L. Zhang. Schematizing Trust in Named Data Networking. In ICN'15, 2015

Flow application: trust relationship

- Example trust relationship

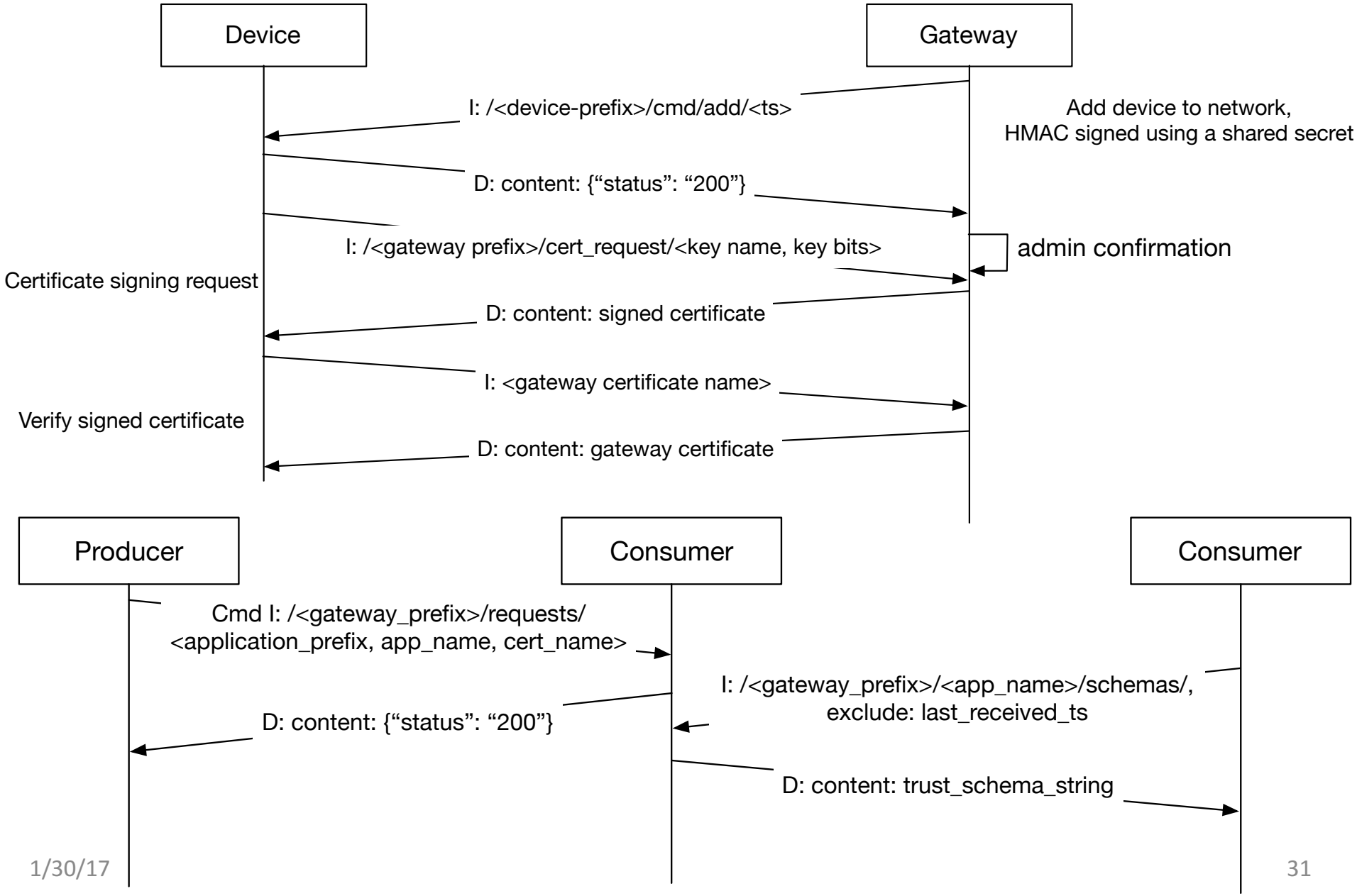


NDN-IoT: bootstrap

- Add device to home network
  - This step establishes trust relationship between gateway and the added device
  - Follows the process introduced in NDN-pi [4]: bootstrap trust between the gateway and a device using a shared secret established offline
- Setting the device up as a producer / consumer
  - This step establishes trust relationship between device identity and application data it produces
  - To work as a producer, the device sends a command interest to the gateway asking for permission to publish under an application prefix
  - To work as a consumer, the device sends interests for the trust schema of an application

[4] A. Bannis, J. Burke. Creating A Secure, Integrated Home Network of Things with Named Data Networking, Tech. Rep. NDN-0035, NDN, 2015

NDN-IoT: bootstrap sequence



NDN-IoT: discovery

- The framework implements distributed discovery (similar to ChronoSync [2]) of devices in a multicast namespace:

  /home/devices/discovery/<digest>

  – Digest is a hash of names of devices known by the producer; Data content is the list of names

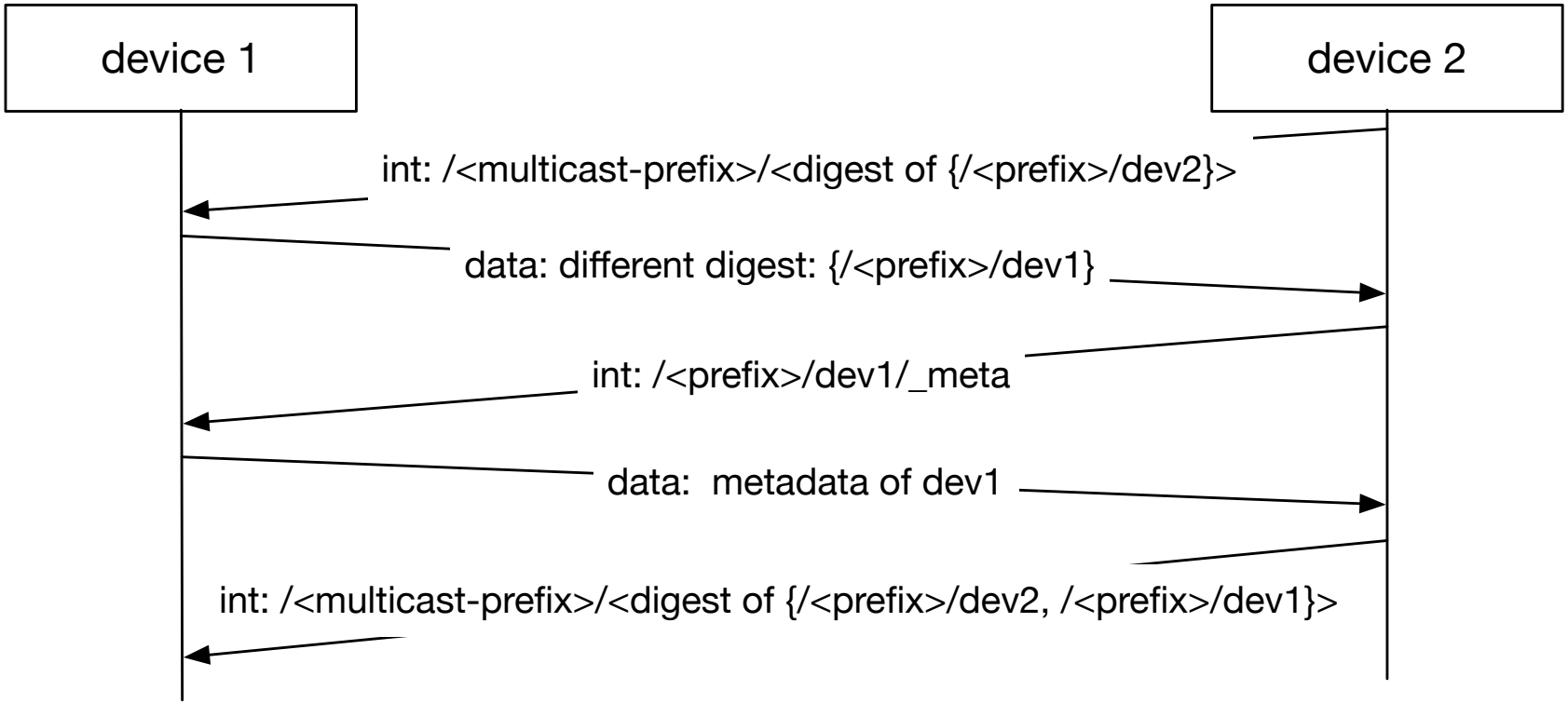
  – Each participant adds its own device name to this list and reply to interests for old digests

  – Upon receiving data, the participant issues device interest to know what application data the device publishes

- Extensions to multi-homed applications interesting future work.

[2] Z. Zhu and A. Afanasyev. Let's ChronoSync: Decentralized dataset state synchronization in NDN. In ICNP, 2013.

NDN-IoT: discovery sequence

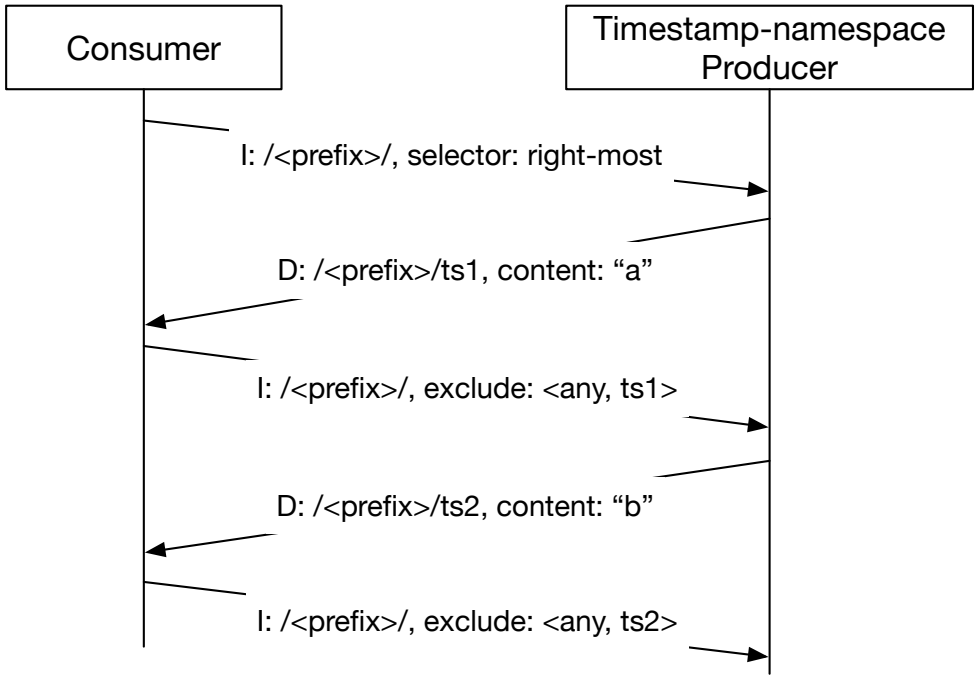


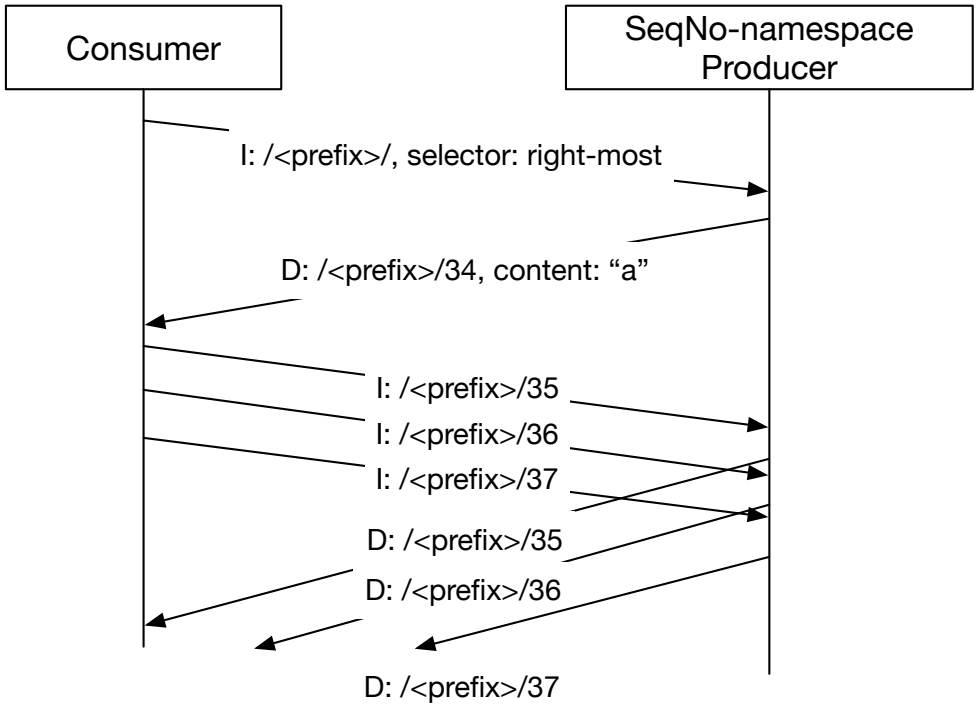
NDN-IoT: app-level pub/sub

- Pub/sub is a common communication paradigm in IoT applications
- The framework implements pull-based pub/sub under two types of namespace abstractions
  - Timestamp-based namespace: uses exclusion filter to repeatedly ask for content
  - Sequence-number-based namespace: pipelines interests for the next few sequence numbers

NDN-IoT: app-level pub/sub sequence

- Interest-data exchanges for app-level pub/sub in timestamp and sequence-number namespaces:



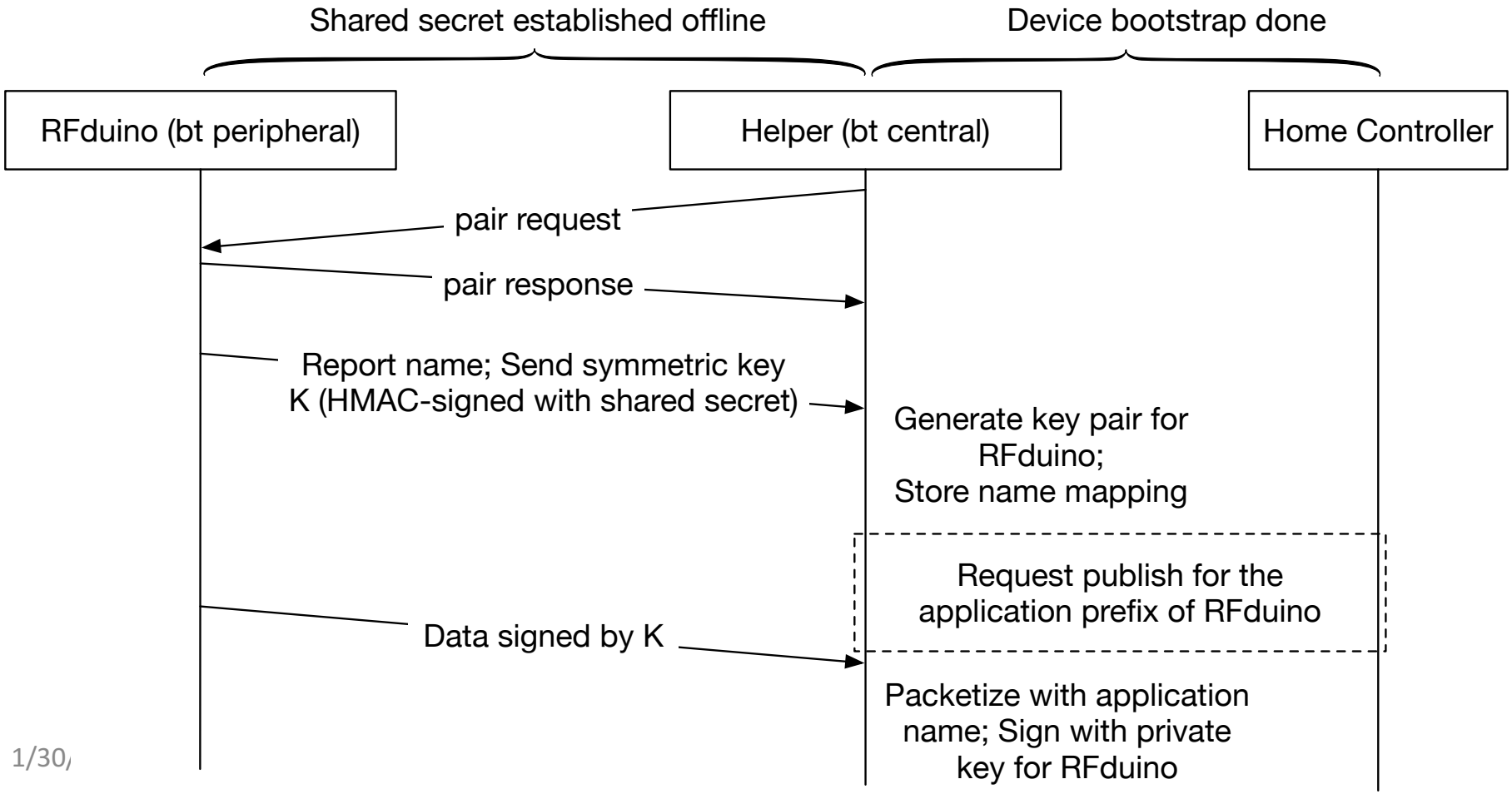


NDN-IoT: constrained devices

- The framework supports constrained devices by pairing them with more powerful “helpers”
  - Constrained devices may not run the forwarder, or do asymmetric signing, and instead delegates signing to the helper
  - We assume each constrained device has established a shared secret with a helper offline

Flow: constrained device authorization

- To make data published by constrained devices (e.g. an RFduino) verifiable, the device and its helper go through the following sequence



“Flow” Application

Flow: introduction

- Flow is a home IoT game application built on top of NDN-IoT framework. Players
  - Interact with the game by walking around in an area tracked by OpenPTrack (opt)
  - Sees physical tracks affect the terrain in a virtual space rendered by Unity3D game engine
  - Drop social media images to the virtual space using a mobile webpage
  - Collectively control the angles the virtual cameras are facing by rotating gyroscopes

Flow: components

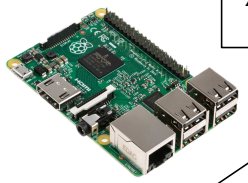
Component	Device	Role
OpenPTrack (opt)	Linux workstation machine	Produces time-series data (location of multiple tracked persons) at 30Hz
Mobile website	Mobile phone	Sends environment control commands and consumes environment status
Virtual camera control	RFduino	Produces NDN data from gyroscope reading. Packetize, and send to helper (RaspberryPi) via bluetooth
Controller/Gateway	RaspberryPi2	Serves as the trust anchor (runs NDN-pi controller)
Unity3D game engine	OSX machine	Consumes opt, phones, and Arduino data; and renders the virtual environment

Flow: app message exchanges

RFduino1/Gyro
(ble peripheral)

Packetized NDN data with HMAC signature

OpenPTrack Publisher box
Application prefix: /home/flow1/opt1
Device name: /home/devices/ubuntu/123



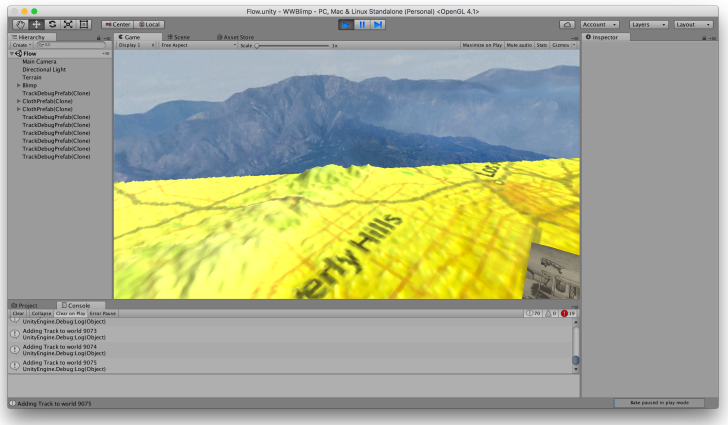
RaspPi2 (ble central)
Application prefix: /home/flow1/gyros/gyro1
Device prefix: /home/devices/rpi2/456

1A. Unity asks for a "track_hint"

1B. names when asking for a track location

3A. Unity asks for gyroscope data, produced by RaspberryPi helper on behalf of RFduinos connected via Bluetooth LE

Unity Laptop
Application prefix: /home/flow1/unity1/
Device name: /home/devices/osx/234



2A. Mobile phone sends command interest and initiates "track matching" of phone's ID to Track ID provided by OPT

2B. command interest to control the dropping of images in Unity

User Mobile Phone
Device name: /home/devices/phone/345

Please refer to next page for diagram key

Flow: app message exchange keys

Diagram Key for NDN Interest-Data Exchange

1A.

Interest: /home/flow1/opt1/<run_id>/hints/, exclude: <last_received_timestamp>
Data: Interest name + <timestamp>, content: {“id”: 45, “seq#”: 312}

1B.

Interest: /home/flow1/opt1/<run_id>/tracks/<track_id>/<seq#>
Data: Interest name, content: {“x”: 1.0, “y”: 0.9, “z”: -0.3}

2A.

Command interest: /home/flow1/unity1/<action:match, id:alice_phone>
Data: Interest name, content: {“status”: “200”, “data”: “<html>track 45, show links</html>“}

2B.

Command Interest: /home/flow1/unity1/<action:link_click, id:alice_phone, link:img_3>
Data: interest name, content: {“status”: “200”}

3A.

Interest: /home/flow1/gyros/gyro1/, exclude: <last_received_timestamp>
Data: Interest name, content: {“p”: 0.3, “y”: 0.5, “r”: 0.1}

Scenario 1: adding gateways and new devices to home

- User has several gateways (e.g. routers) in his home; each one may cover several rooms
- When installing the trust anchor gateway at home for the first time, user will be asked to provide its name
  - For example, router serial-1234 should generate a key pair for identity "/home/devices/gateway/", and a self-signed certificate)
- The user could then add devices trusted by the trust anchor gateway (or by other gateways who are already added using the process "bootstrap – add device". In this case, multiple gateways form a chain of trust.)

Scenario 2: buying Flow and installing at home

- After acquiring the necessary hardware and setting them up at home, the user buys Flow software, verifies the binaries with developer's certificate, and installs on various devices
- Before launching, user configures Flow application prefixes on each piece of hardware, and upon first launch the application binary requests producing authorization.
- Installation is complete after user authorizes application publishing requests.

Scenario 3: replacing a device using the same application level name

- Imagine that an old “gyro1” breaks, we replace it with a new one also called “gyro1”
- To achieve this, the user follows the application prefix configuration step described in previous slide
  - Name the new gyroscope “gyro1”
  - Run “bootstrap – producer authorization” on the new gyroscope to tell the gateway to update trust schema with new device certificate for data under “gyro1” prefix

Uncovered functionality from IoTDI '16

- Section VI.D (access control)
  - Not implemented as its functionality is mostly covered in NDN-CCL
  - This functionality is available in PyNDN2, ndn-js, jndn, ndn-cpp, as well in ndn-group-encrypt
- Section VI.E (aggregation)
  - Not implemented as not immediately useful for our current application
  - This is implemented in mini-BMS, NDN Building Management System emulation in mini-ndn

Future work: NDN-IoT

- Library interface
  - Interface design needs improvement (e.g. not straight forward return values and parameters in bootstrap and discovery; not conformed OnValidationFailed interface)
- Library
  - Consistent error reporting and exception handling
  - Build system updates
  - Example code beyond the Flow demo
- Demonstration
  - Illustrate discovery
  - Illustrate verified data vs unverified data

Future work: Application

- Aesthetic / functionality update
  - Expect to provide an update later this quarter
- Multi-user support
  - Expand multi-user support / web bootstrapping
- OpenPTrack message update
  - Long-term prospect
  - Secure, name-based messaging for the Robot Operating System (ROS)

Links and documentation

- Code repository
  - NDN-IoT framework
    - Functionality overview
    - Interface description
  - Flow application
- Technical guide (installation and troubleshooting)
- Demo poster

Other NDN IoT Projects

- **NDN-BMS**: encryption-based access control
  - Wentao Shang, Qiuhan Ding, Alessandro Marianantoni, Jeff Burke, Lixia Zhang. "**Securing Building Management Systems Using Named Data Networking.**" In IEEE Network, Vol. 28, no. 3, May 2014.
- **NDN-ACE**: authorization framework for actuation apps
  - W. Shang, Y. Yu, T. Liang, B. Zhang, and L. Zhang, "NDN-ACE: Access Control for Constrained Environments over Named Data Networking," NDN Project, Tech. Rep. NDN-0036, Revision 1, December 2015.

- **NDN-PI**: toolkit for NDN dev on Raspberry Pi
  - https://github.com/remap/ndn-pi
- **NDN on Arduino**: minimal app for Arduino
  - https://github.com/ndncomm/ndn-btle
- **RIOT OS**: the friendly OS for IoT
  - https://www.riot-os.org/
  - http://irl.cs.ucla.edu/~wentao/ndn-riot-os-poster.pdf
    - NDN on RIOT coming soon

Conclusion

- NDN names IoT data directly and retrieves data based on those names
- Data-centric security enables fine-grained access control with least privilege
- NDN efficiently supports communications over various L2 technologies and network environments
- Frameworks can be built on top of L3 primitives to facilitate application development

References

- [1] W. Shang, A. Bannis, T. Liang, Z. Wang, Y. Yu, A. Afanasyev, J. Thompson, J. Burke, B. Zhang, L. Zhang. Named Data Networking of Things (Invited Paper). In IoTDI, 2016
- [2] Z. Zhu and A. Afanasyev. Let's ChronoSync: Decentralized dataset state synchronization in NDN. In ICNP, 2013.
- [3] Y. Yu, A. Afanasyev, D. Clark, K. Claffy, V. Jacobson, and L. Zhang. Schematizing Trust in Named Data Networking. In ICN'15, 2015
- [4] A. Bannis, J. Burke. Creating A Secure, Integrated Home Network of Things with Named Data Networking, Tech. Rep. NDN-0035, NDN, 2015