Breaking out of the cloud: Local trust management and rendezvous in Named Data Networking of Things

Wentao Shang, Zhehao Wang, Alexander Afanasyev, Jeff Burke, Lixia Zhang

UCLA

2017 ACM/IEEE Int. Conf. on Internet-of-Things Design and Implementation

“As we were finalizing this paper, a benign operational error brought down Amazon cloud service and impacted *a large number* of IoT services as well as other applications. This incident reminds us that cloud services are not immune to failures, further underscoring the value of this work”

Summary of the Amazon S3 Service Disruption in the Northern Virginia (US-EAST-1) Region

We'd like to give you some additional information about the service disruption that occurred in the Northern Virginia (US-EAST-1) Region on the morning of F **Feb 28, 2017** The Amazon Simple Storage Service (S3) team was debugging an issue causing the S3 billing system to progress more slowly than expected. At 9:37AM PST,

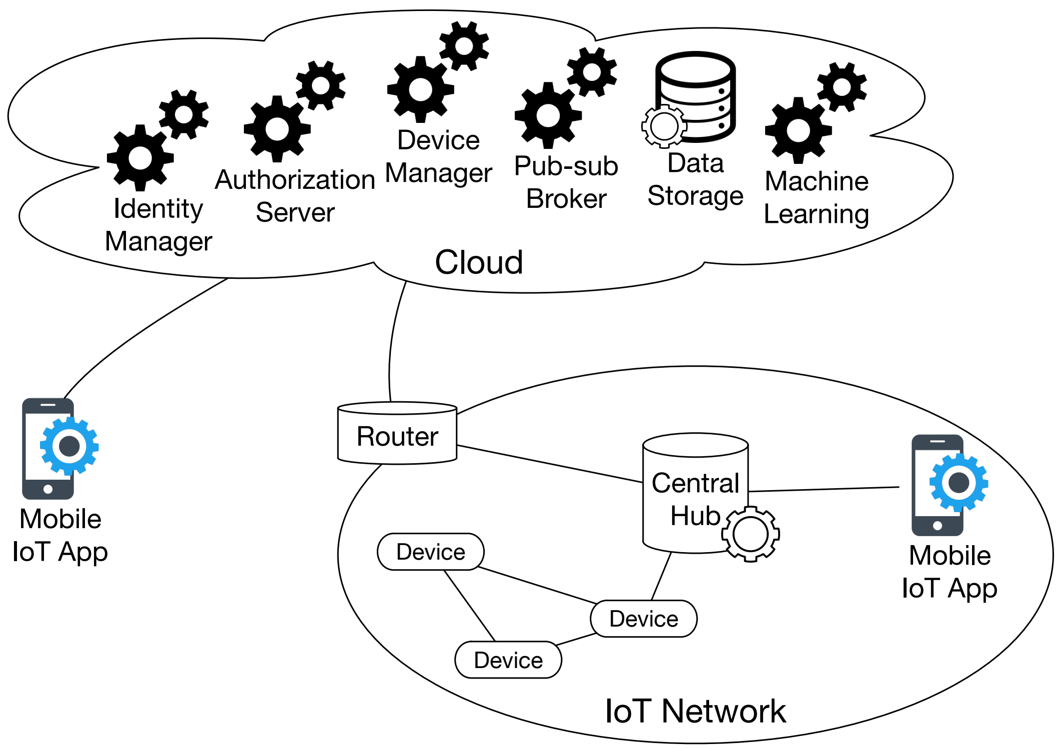
One of the inputs to the command was entered incorrectly and a larger set of servers was remove than intended.

These servers supported two other S3 subsystems

one of them, the index subsystem, manages the metadata and location information of all S3 objects in the region....

properly to correctly operate. The placement subsystem is used during PUT requests to allocate storage for new objects. Removing a significant portion of the capacity caused each of these systems to require a full restart. While these subsystems were being restarted, S3 was unable to service requests. Other AWS services in the US-EAST-1 Region that rely on S3 for storage, including the S3 console, Amazon Elastic Compute Cloud (EC2) new instance launches, Amazon Elastic Block Store (EBS) volumes (when data was needed from a S3 snapshot), and AWS

Cloud-centric (Silo) Approach to IoT



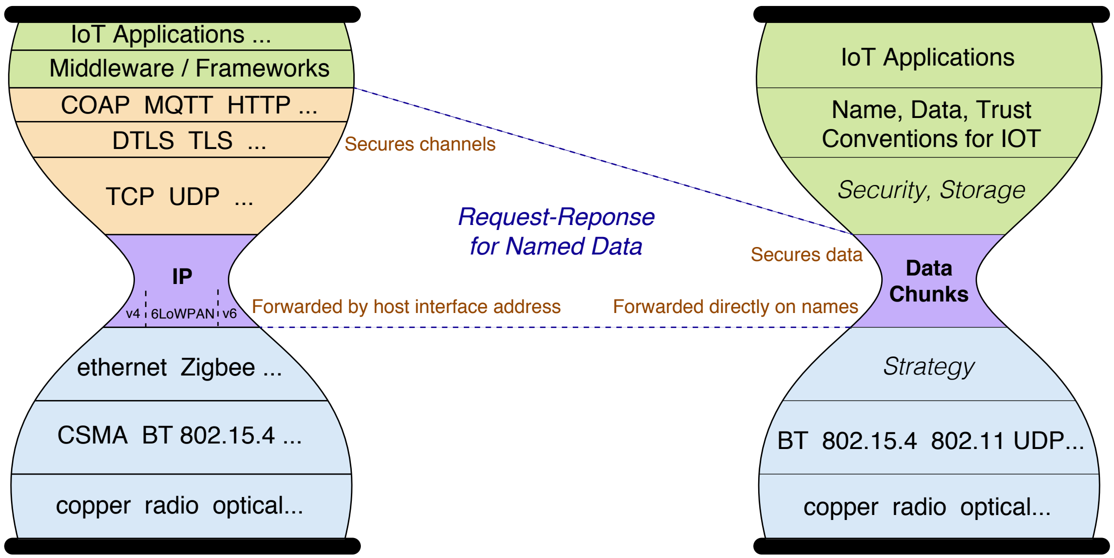
Some of them started delegating certain basic functions locally, after configuration phase

So, the existing cloud-centric IoT systems

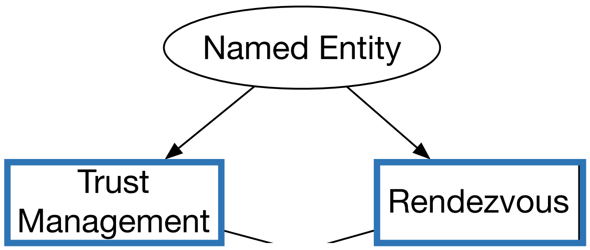
- Use remote services to achieve local functions
  - Add/remote devices and users
  - Discover, authenticate & authorize access to local data and services
- Can hurt real-time interactive experience when local interactions have to go through cloud
- Subject local IoT operations to remote failures
  - Public Internet connectivity to the cloud may be lost
  - Cloud services are not immune to failures, either
- Expose private data to cloud providers

Named Data Networking of Things (IoTDI'16)

- Name-based network forwarding
- Interest-Data exchange model matches REST API
- Data-centric security



Rethink the IoT Service Architecture



Example: AWS IoT platform

Amazon Resource Name (ARN) of my living room lamp:
arn:aws:iot:us-west-1:wentao:things/LivingRoomLamp

Enabling Cloud-independent IoT by NDN

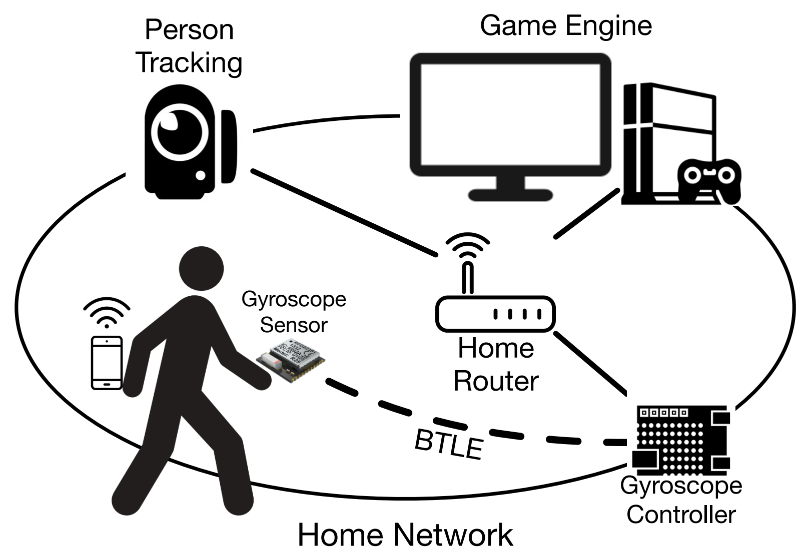
- **Named entities**: name "things" *within local context*
- **Trust management**: define the relation between data names and signing key names, *within local context*
- **Rendezvous**: publish & synchronize application names under a local discovery namespace, *within local network/context*

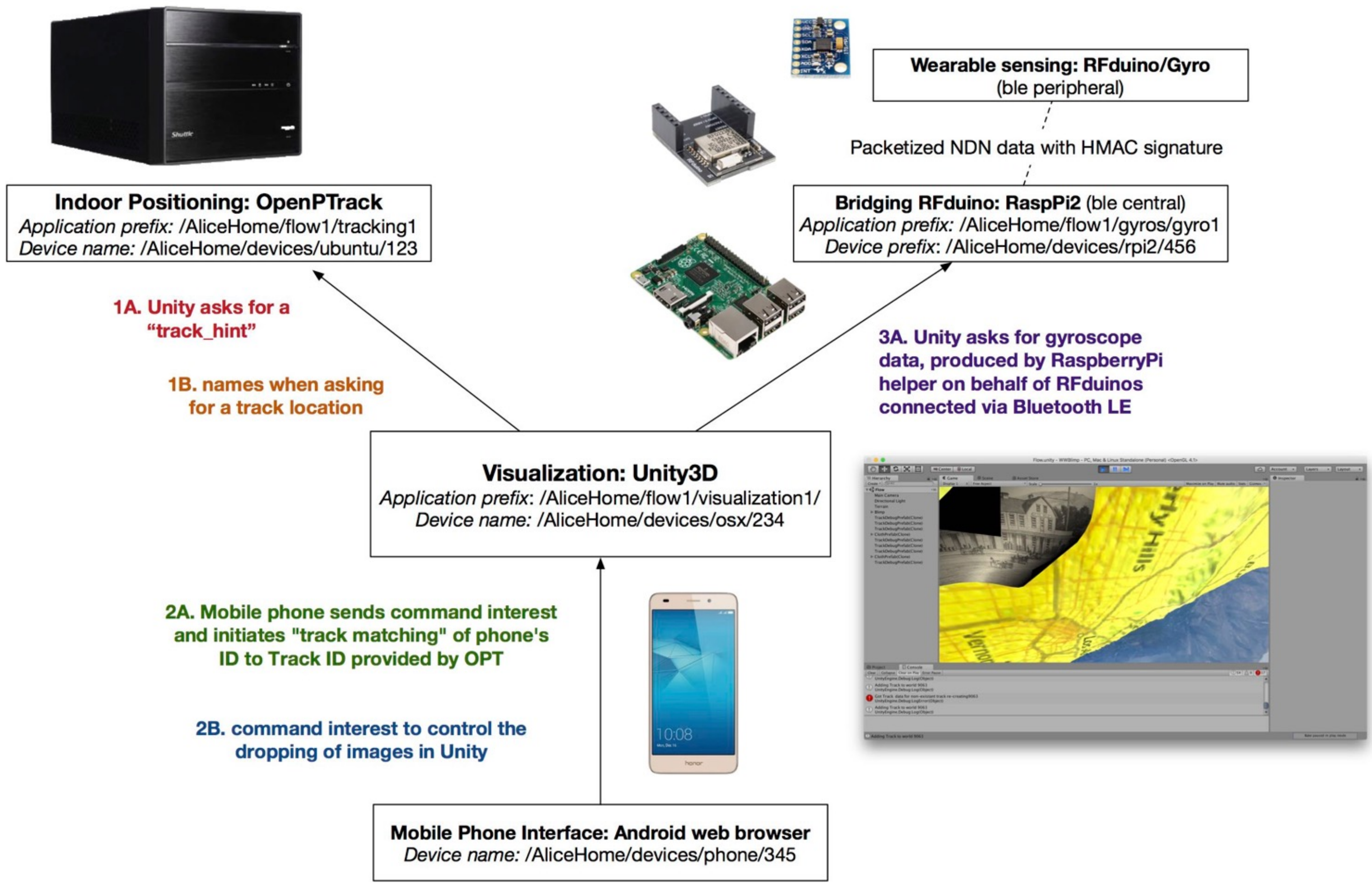
1. Other services/applications can be bootstrapped from the above
2. Cloud services, whenever available, can provide functions beyond the capability of local IoT systems
   - e.g., voice recognition, data analytics, long term storage, etc

Experimentation: Flow – a cloud-independent home entertainment system over NDN

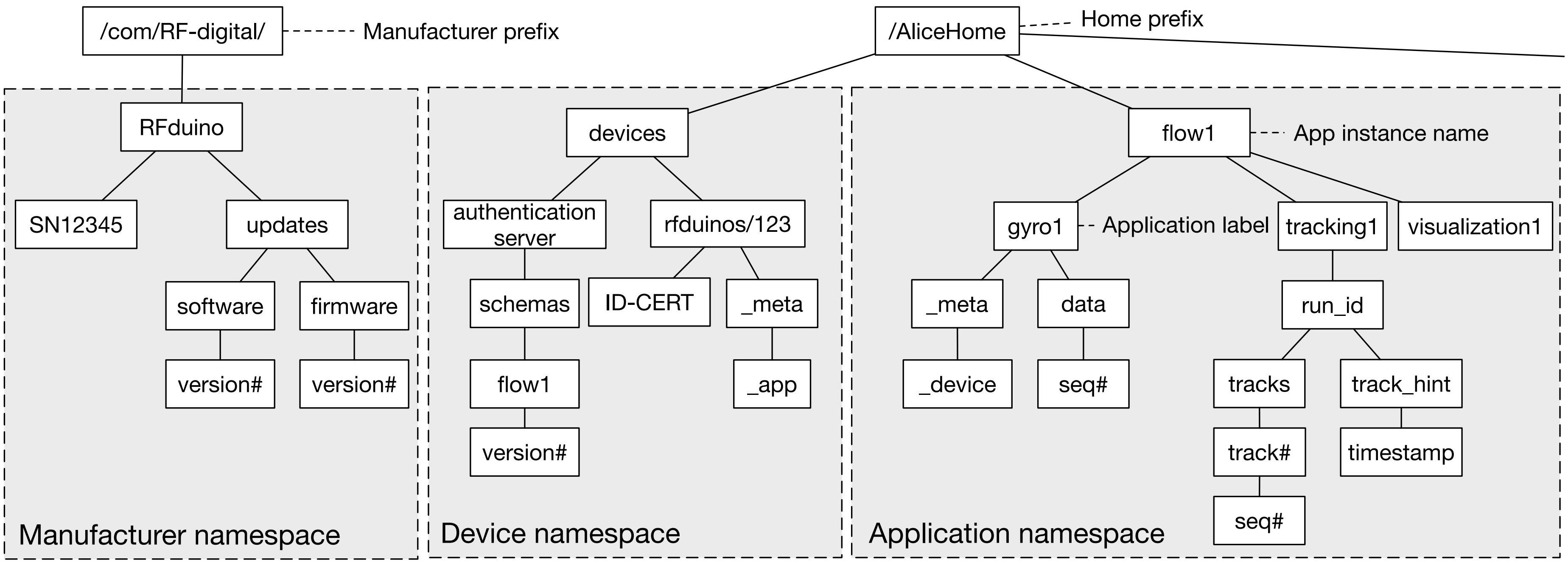
A multi-user “exploration game” prototype, utilizing:

- *Indoor positioning*: player’s physical position modifies virtual game world
- *Wearable sensing*: player wears gyroscope to control orientation of virtual camera
- *Mobile phone interface*: used by player to control actions in virtual game world
- *Game engine*: visualization rendered in Unity.



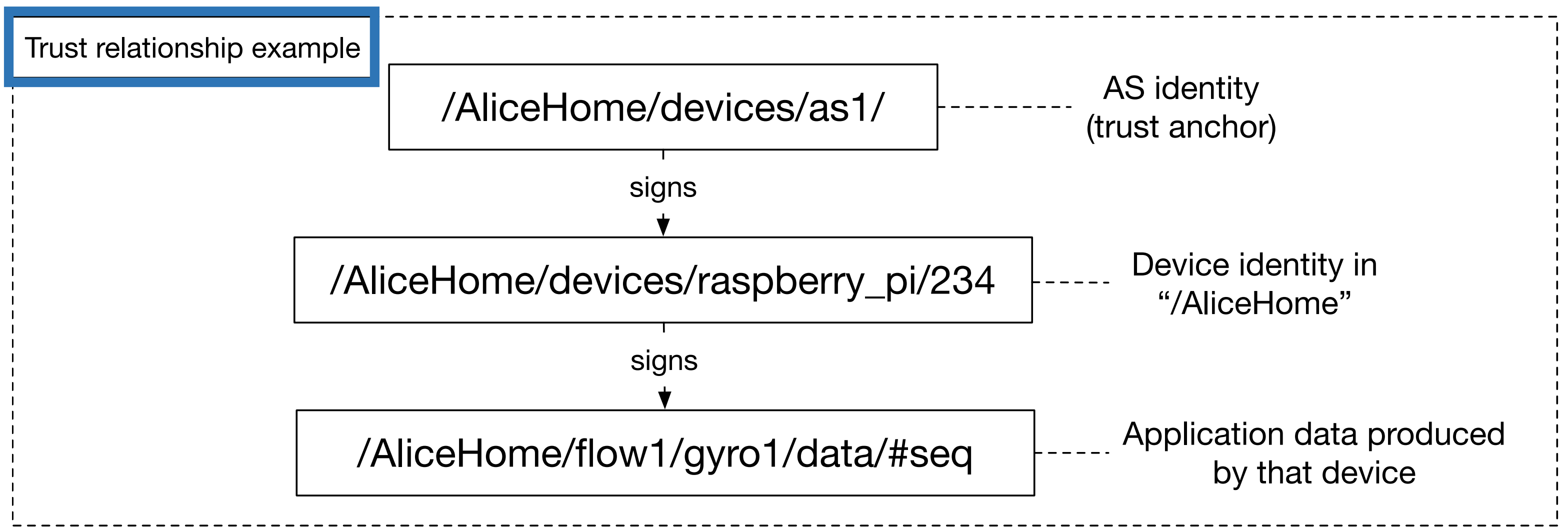

Indoor Positioning: OpenPTrack
Application prefix: /AliceHome/flow1/tracking1
Device name: /AliceHome/devices/ubuntu/123
Wearable sensing: RFduino/Gyro
(ble peripheral)
Packetized NDN data with HMAC signature
Bridging RFduino: RaspPi2 (ble central)
Application prefix: /AliceHome/flow1/gyros/gyro1
Device prefix: /AliceHome/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
Visualization: Unity3D
Application prefix: /AliceHome/flow1/visualization1/
Device name: /AliceHome/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
Mobile Phone Interface: Android web browser
Device name: /AliceHome/devices/phone/345

Naming and Identity

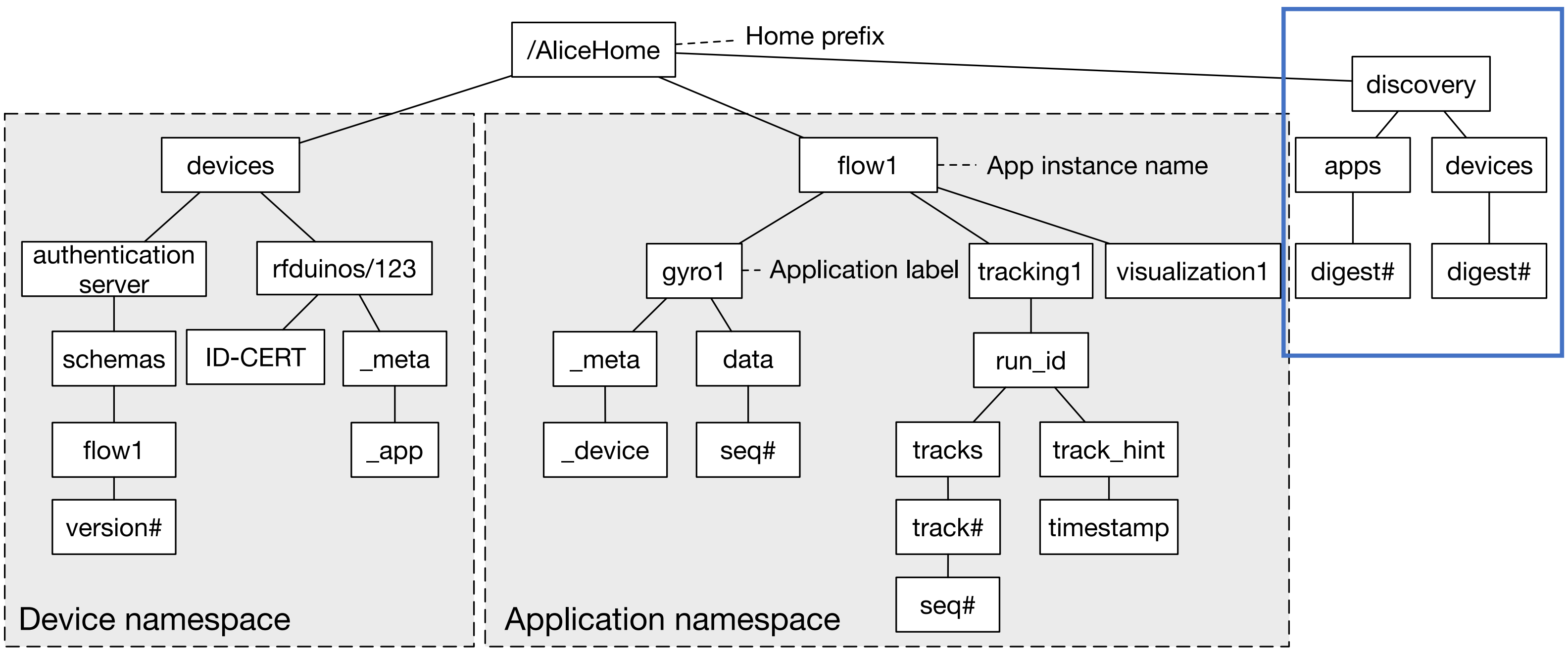


Trust Management

AS: Authentication server

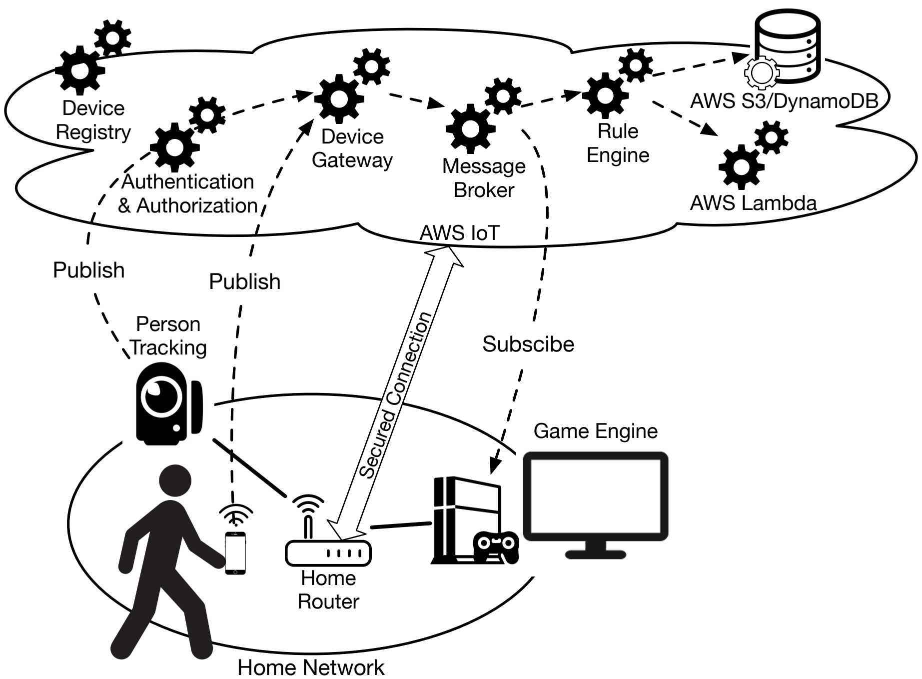


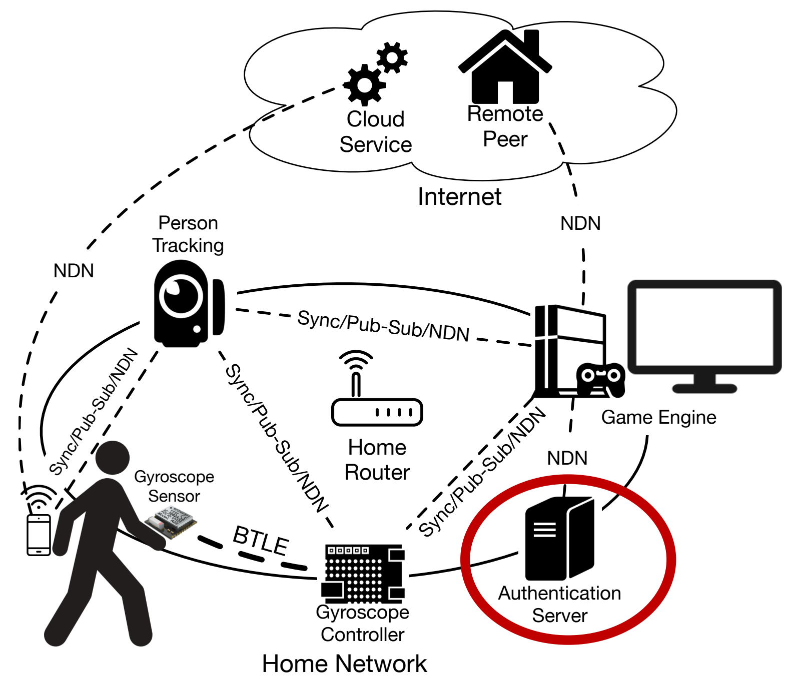
Rendezvous



/AliceHome
Home prefix
discovery
apps
devices
digest#
digest#
devices
authentication server
rfduinos/123
schemas
ID-CERT
_meta
flow1
_app
version#
Device namespace
flow1
App instance name
gyro1
Application label
tracking1
visualization1
_meta
data
_device
seq#
run_id
tracks
track_hint
track#
timestamp
seq#
Application namespace

Architecture: AWS-IoT vs. NDN-IoT





Semantic maning → local trust anchor → local autonomy

Implementation

- Indoor positioning with OpenPTrack over NDN
  - Publish position data at 30Hz and metadata at lower rate
- Wearable sensing with RFduino22301 and gyroscope MPU6050
  - RFduino generates NDN data at 2Hz and transfers to a Raspberry Pi 2 controller for signing and publishing
- Mobile interface on Android phone using NDN.JS library
  - Generates command Interests to control virtual environment and update player's position
- Visualization with Unity3D game engine
  - Consumes positioning and gyro data, and receives command Interests

Links to code

- Code repository (https://github.com/remap/ndn-flow)
  - NDN-IoT framework (https://github.com/remap/ndn-flow/tree/master/framework)
    - Functionality overview
    - Interface description
  - Flow application (https://github.com/remap/ndn-flow/tree/master/application)
- Technical guide (installation and troubleshooting)
- Demo poster
- Application screen recording

Takeaway

- Existing cloud-centric IoT systems: developed along a path of least resistance
  - Readily available point-to-point TCP/IP communication infrastructure
  - Leveraging economy of scale
- Cloud dependency leads to a number of problems
  - Service availability, privacy, end user autonomy, and more.
- This work: exploring NDN in enabling cloud-independent IoT systems

  Application-defined data naming within local context →
  - Discovering local components and services through local sync
  - Expressing trust relations by names with schematized trust
  - Securing and exchanging named data at network layer

https://named-data.net/publications/ndn-breaking-out-of-cloud-iotdi-2017/