NDN: NP Network Environments Update

Jeff Burke
NSF FIA PI Meeting
November 19, 2015

Collaborative work by the NDN team.

NDNcomm 2015 Hackathon

NDN-NP NetEnv Progress to Date

Where we started –

- **Incorporate security**: Name-based trust definition, verification, confidentiality.
- **Practical deployment** needs that yield research challenges: publisher mobility, autoconfig, trust bootstrapping, etc.
- **Higher-level communication concepts** – e.g., sync, manifests.

What also emerged –

- **App / strategy relationships**.
- **More / deeper namespace tussles.**
- **Traffic measurement requirements**. (Not in these slides)
- **Performance requirements**. (Finally!)

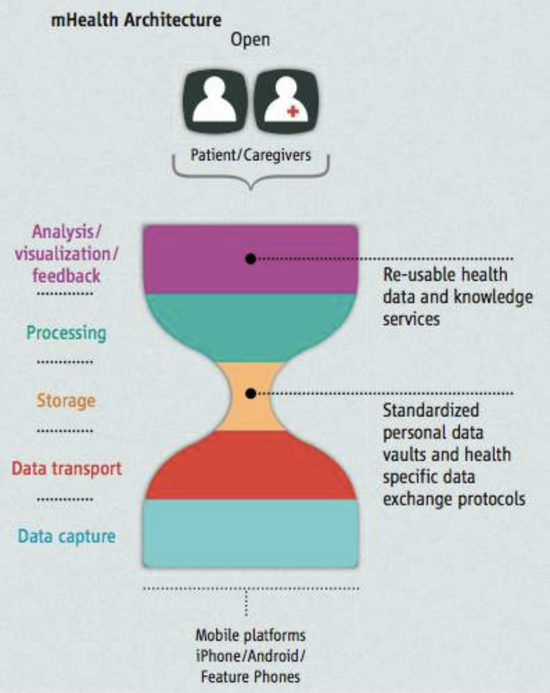
> "It is widely accepted that creative design is not a matter of first fixing the problem and then searching for a satisfactory solution concept; instead it seems more to be a matter of **developing and refining together both the formulation of the problem and ideas for its solution…**"
>
> - Cross & Dorst (1999), quoted by Brooks (2010)

NDNFit

Haitao Zhang, Alexander Afanasyev, Jianxun Cao, Euihyun Jung, Jiewen Tan, Jeff Thompson, Yingdi Yu, Jeff Burke, Dan Pei, Christian Tschudin, Lixia Zhang, and others.

NDNFit: Open mHealth example application

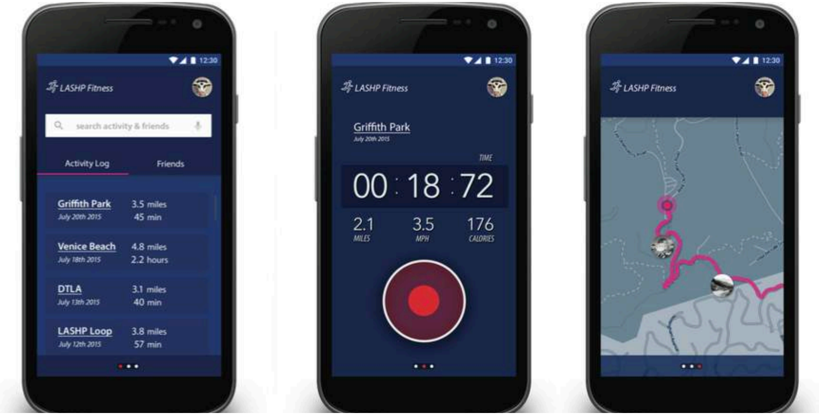
An ecosystem conceived with data exchange as the thin waist (Sim & Estrin, 2010), which is natural for NDN.


mHealth Architecture
Open
Patient/Caregivers
Analysis/ visualization/ feedback
Processing
Storage
Data transport
Data capture
Re-usable health data and knowledge services
Standardized personal data vaults and health specific data exchange protocols
Mobile platforms iPhone/Android/ Feature Phones

How do we conceive and build a familiar-looking application that demonstrates interaction in this ecosystem?

NDN enables a user-centric reformulation of health and wellness data management.

- **User owns the data and controls the root of trust**
- **Ecosystem of interoperable applications using named data as API**



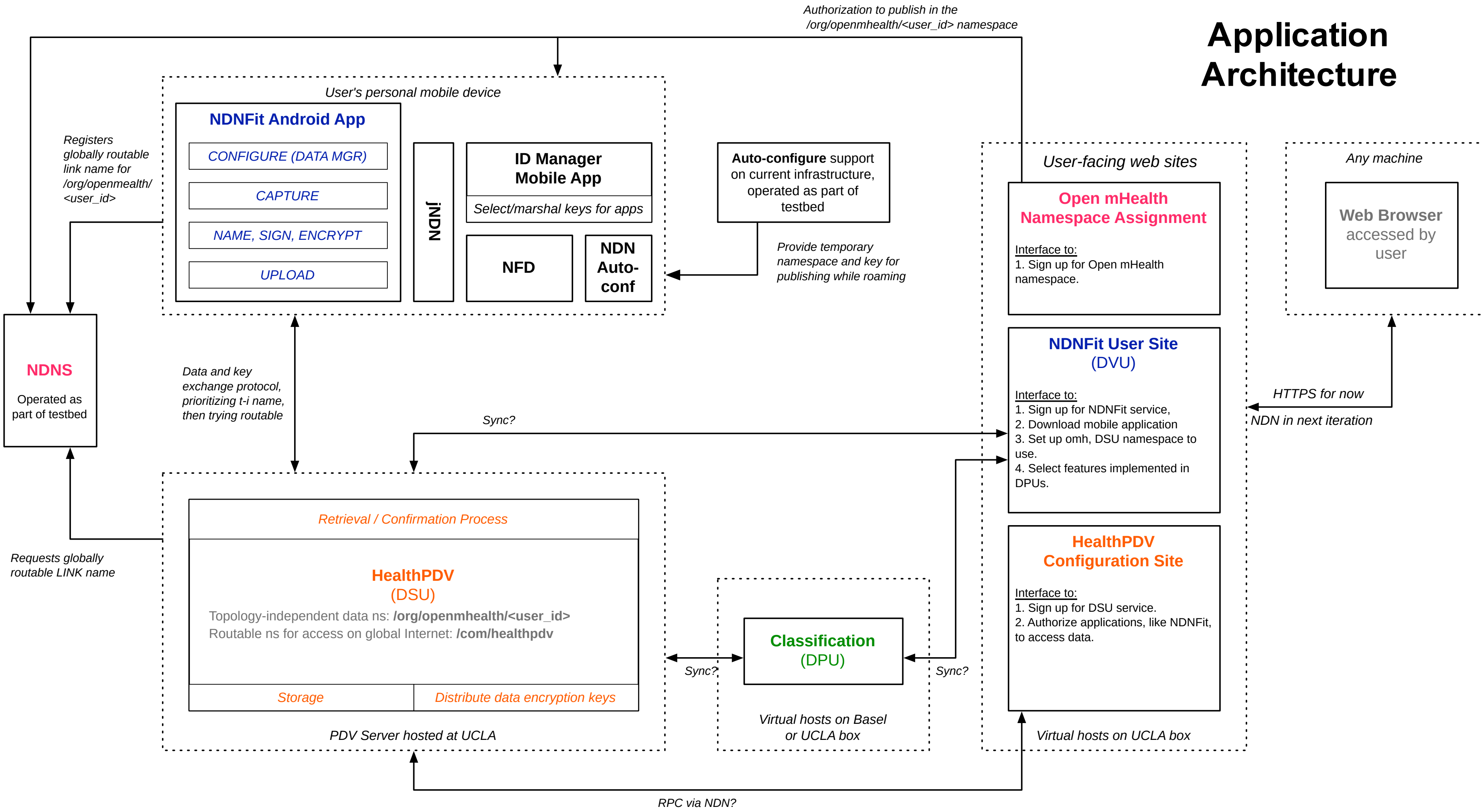
Griffith Park 3.5 miles
July 20th 2015 45 min

Venice Beach 4.8 miles
July 18th 2015 2.2 hours

DTLA 3.1 miles
July 13th 2015 40 min

LASHP Loop 3.8 miles
July 12th 2015 57 min

Application Architecture



Authorization to publish in the /org/openmhealth/<user_id> namespace

User's personal mobile device

NDNFit Android App

CONFIGURE (DATA MGR)

CAPTURE

NAME, SIGN, ENCRYPT

UPLOAD

jNDN

ID Manager Mobile App

Select/marshal keys for apps

NFD

NDN Auto-conf

Auto-configure support on current infrastructure, operated as part of testbed

Provide temporary namespace and key for publishing while roaming

Registers globally routable link name for /org/openmealth/<user_id>

NDNS

Operated as part of testbed

Data and key exchange protocol, prioritizing t-i name, then trying routable

Sync?

Requests globally routable LINK name

Retrieval / Confirmation Process

HealthPDV

(DSU)

Topology-independent data ns: **/org/openmhealth/<user_id>**

Routable ns for access on global Internet: **/com/healthpdv**

Storage

Distribute data encryption keys

PDV Server hosted at UCLA

Sync?

Classification

(DPU)

Sync?

Virtual hosts on Basel or UCLA box

RPC via NDN?

User-facing web sites

Open mHealth Namespace Assignment

Interface to:

1. Sign up for Open mHealth namespace.

NDNFit User Site

(DVU)

Interface to:

1. Sign up for NDNFit service,
2. Download mobile application
3. Set up omh, DSU namespace to use.
4. Select features implemented in DPUs.

HealthPDV Configuration Site

Interface to:

1. Sign up for DSU service.
2. Authorize applications, like NDNFit, to access data.

Virtual hosts on UCLA box

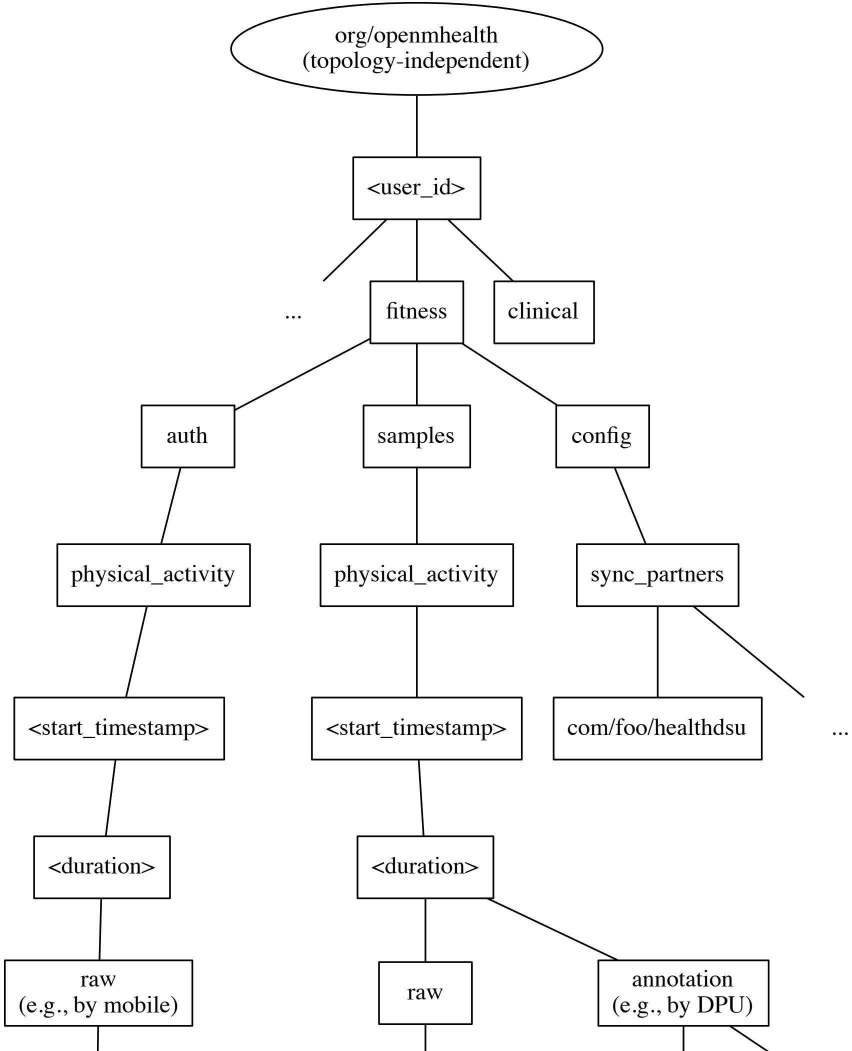
Any machine

Web Browser accessed by user

HTTPS for now

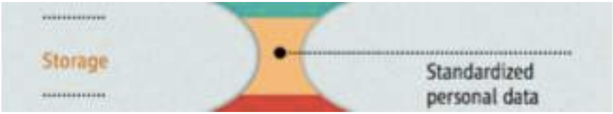
NDN in next iteration

Namespace


org/openmhealth
(topology-independent)
<user_id>
...
fitness
clinical
auth
samples
config
physical_activity
physical_activity
sync_partners
<start_timestamp>
<start_timestamp>
com/foo/healthdsu
...
<duration>
<duration>
raw
(e.g., by mobile)
raw
annotation
(e.g., by DPU)

Data-Centric Security

- Good fit for this application.



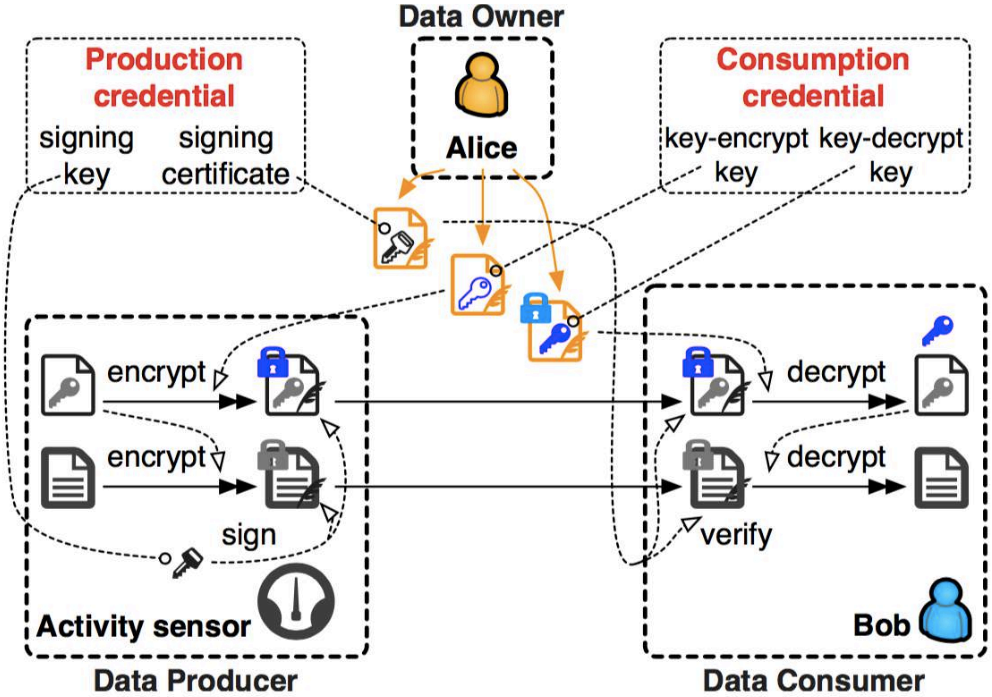
- Reformulates the mapping between architecture and application. (e.g., OAuth is pain point for existing Open mHealth platform because it’s not a great mapping to granular access control by many apps.)

- NDN Approach:
  - Schematized trust (discussed in next net env)
  - Name-based access control (discussed next slide)

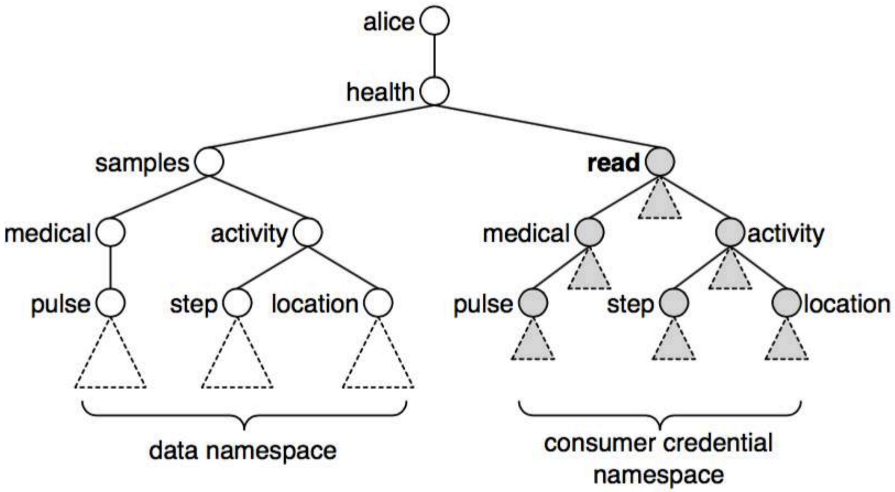
- New pieces / future areas
  - Access control for data-flow style processing.
  - Name confidentiality.

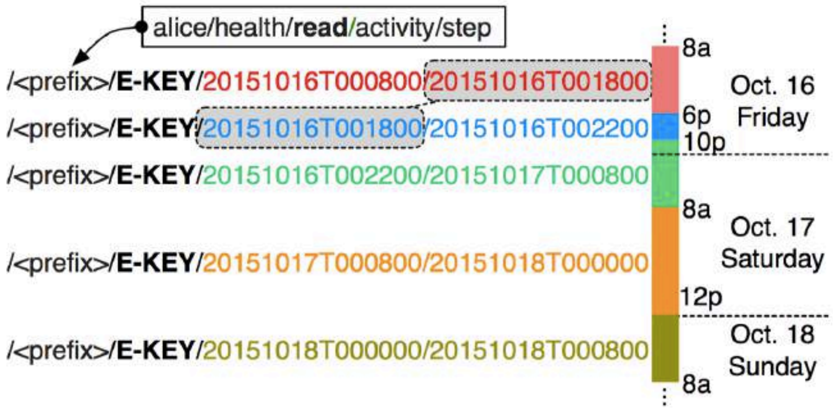
Name-based access control in Open mHealth

- See NDN Tech Report #34, Yu et al., for general approach.
- Initial library support by end of the calendar year.



Name-based access control in Open mHealth




alice/health/read/activity/step
/<prefix>/E-KEY/20151016T000800/20151016T001800
/<prefix>/E-KEY/20151016T001800/20151016T002200
/<prefix>/E-KEY/20151016T002200/20151017T000800
/<prefix>/E-KEY/20151017T000800/20151018T000000
/<prefix>/E-KEY/20151018T000000/20151018T000800
8a
6p
10p
8a
12p
8a
Oct. 16 Friday
Oct. 17 Saturday
Oct. 18 Sunday

ViD Collaboration:
When / how does architecture “bubble up” to users?

NDN Namespaces

- How are users introduced to the application namespaces?
- How much exposure to the namespaces is necessary?

Identity management & Data signing

- How do we get users involved in signing their data?

Managing access control of personal data

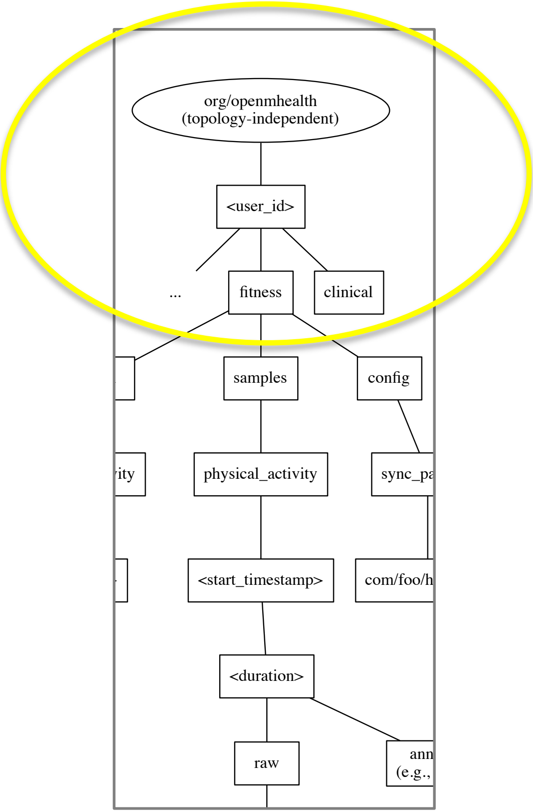
- How do we make data-centric security usable for personal data?

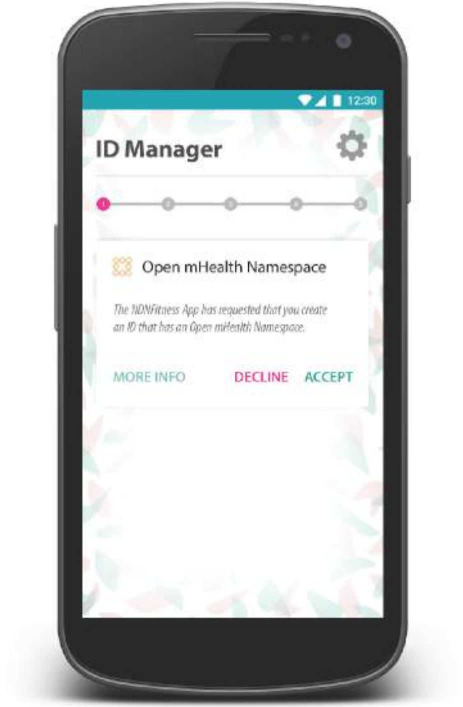
Data-centric Interoperability

- What design choices can be made that move us towards a data-centric ecosystem rather than silo’ed applications?

Work by Dustin O’Hara with UCLA IRL, and others. Supported by NeTS Small with Katie Shilton (UMD).

Selecting Identities: **Identity Manager App**



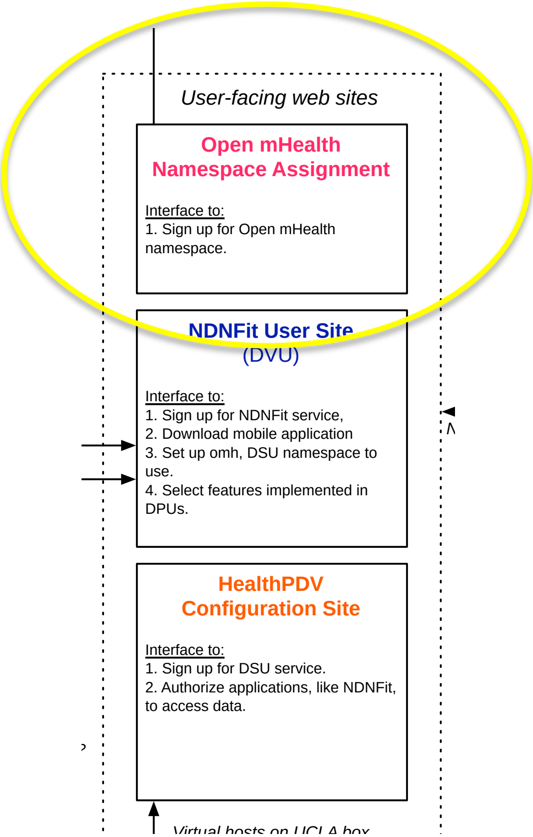


ID Manager

Open mHealth Namespace

The NDNFitness App has requested that you create an ID that has an Open mHealth Namespace.

Configure participation in namespace



User-facing web sites

Open mHealth Namespace Assignment

Interface to:
1. Sign up for Open mHealth namespace.

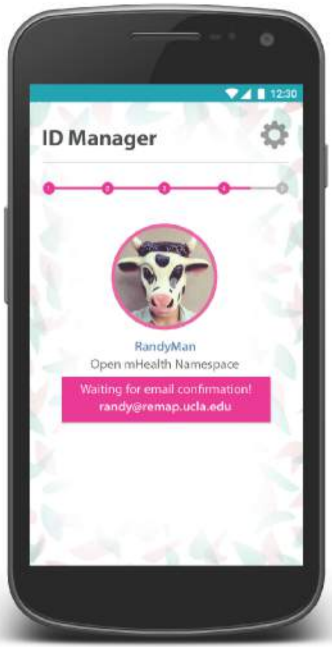
NDNFit User Site
(DVU)

Interface to:
1. Sign up for NDNFit service,
2. Download mobile application
3. Set up omh, DSU namespace to use.
4. Select features implemented in DPUs.

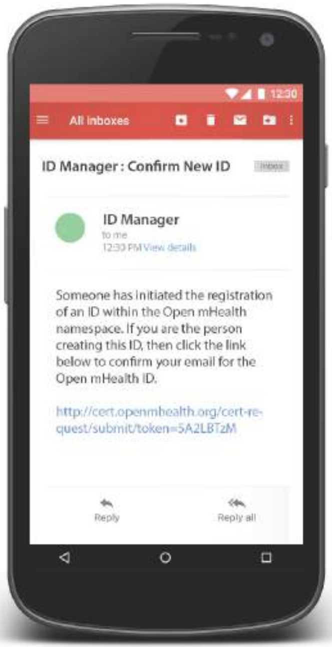
HealthPDV Configuration Site

Interface to:
1. Sign up for DSU service.
2. Authorize applications, like NDNFit, to access data.

Virtual hosts on UCLA box



RandyMan
Open mHealth Namespace
Waiting for email confirmation!
randy@remap.ucla.edu



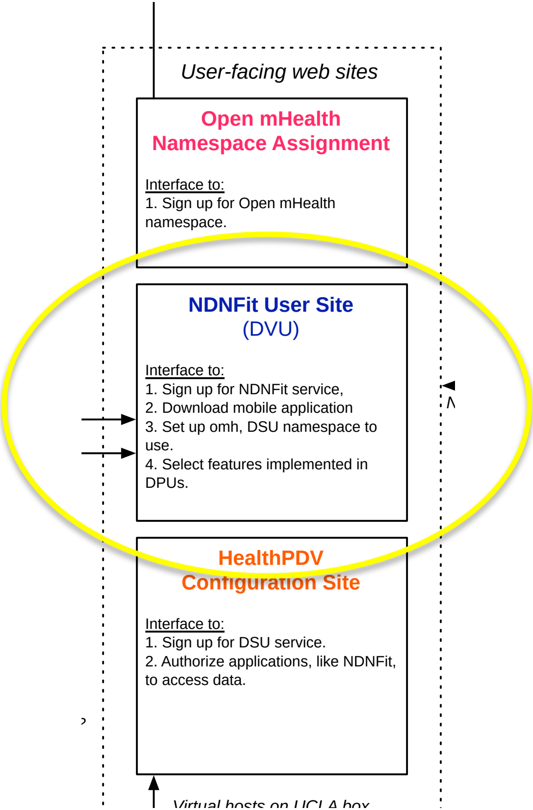
ID Manager : Confirm New ID

ID Manager
to me
12:30 PM View details

Someone has initiated the registration of an ID within the Open mHealth namespace. If you are the person creating this ID, then click the link below to confirm your email for the Open mHealth ID.

http://cert.openmhealth.org/cert-request/submit/token=5A2LBTzM

Create and authorize certificate for this namespace; give a local nickname.



User-facing web sites

Open mHealth Namespace Assignment

Interface to:
1. Sign up for Open mHealth namespace.

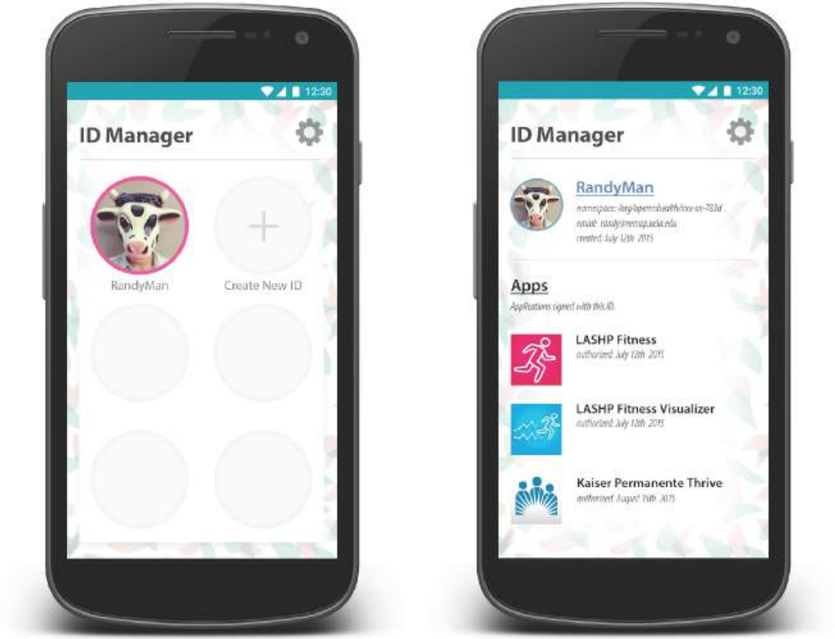
NDNFit User Site (DVU)

Interface to:
1. Sign up for NDNFit service,
2. Download mobile application
3. Set up omh, DSU namespace to use.
4. Select features implemented in DPUs.

HealthPDV Configuration Site

Interface to:
1. Sign up for DSU service.
2. Authorize applications, like NDNFit, to access data.

Virtual hosts on UCLA box



ID Manager

RandyMan
namespace: /org/openmhealth/[illegible]
email: randy@remap.ucla.edu
created: July 12th 2015

Apps
Applications signed with this ID.

LASHP Fitness
authorized: July 12th 2015

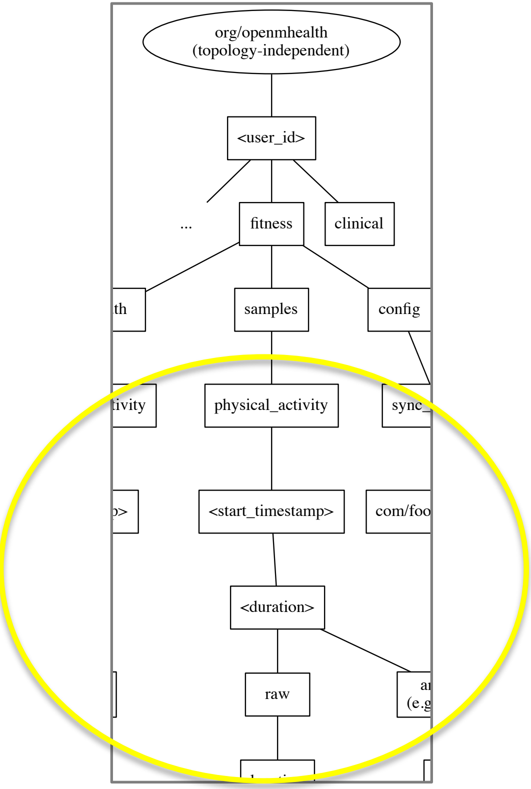
LASHP Fitness Visualizer
authorized: July 12th 2015

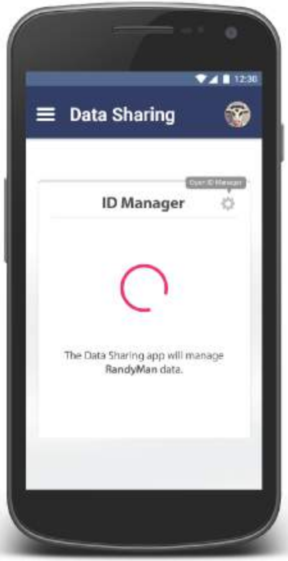
Kaiser Permanente Thrive
authorized: August 15th 2015

Associate with applications

(Open question: How to marshal such certs needed by applications?)

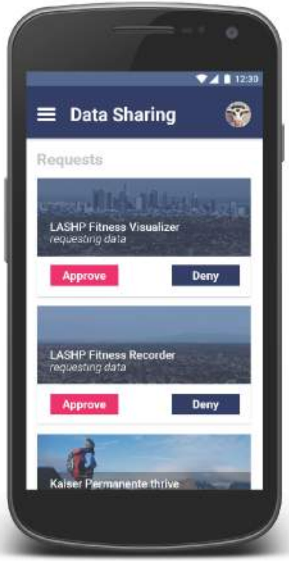
Data Sharing Service





ID Manager

The Data Sharing app will manage RandyMan data.

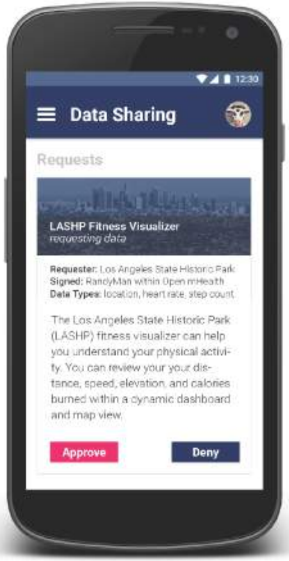


Requests

LASHP Fitness Visualizer
requesting data

LASHP Fitness Recorder
requesting data

Kaiser Permanente thrive



Requests

LASHP Fitness Visualizer
requesting data

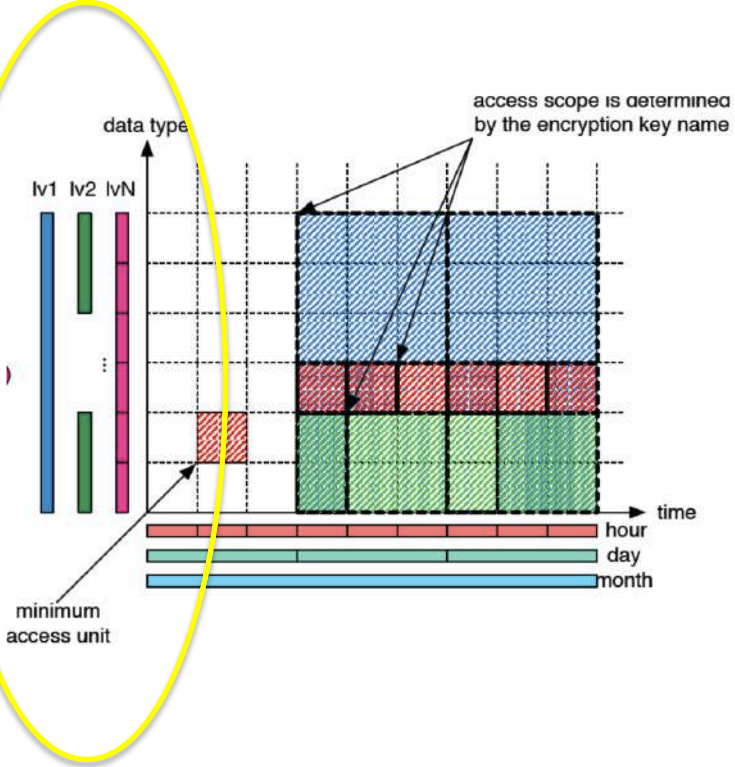
Requester: Los Angeles State Historic Park
Signed: RandyMan within Open mHealth
Data Types: location, heart rate, step count

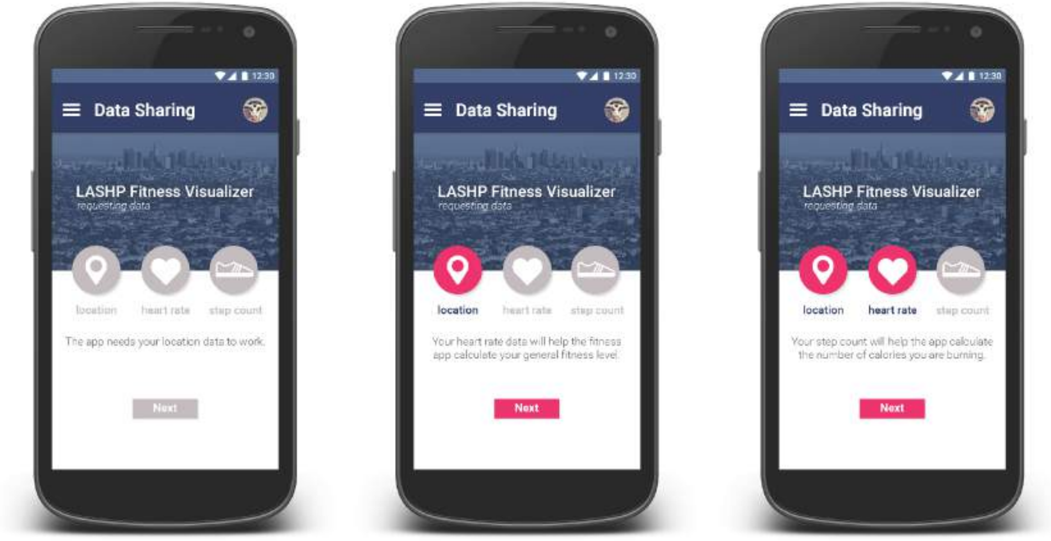
The Los Angeles State Historic Park (LASHP) fitness visualizer can help you understand your physical activity. You can review your your distance, speed, elevation, and calories burned within a dynamic dashboard and map view.

Manage access control constraints on data publication

Model: Selective Sharing (NSF CENS)

TBD: App-specific or common Uis for standard sub-namespaces?



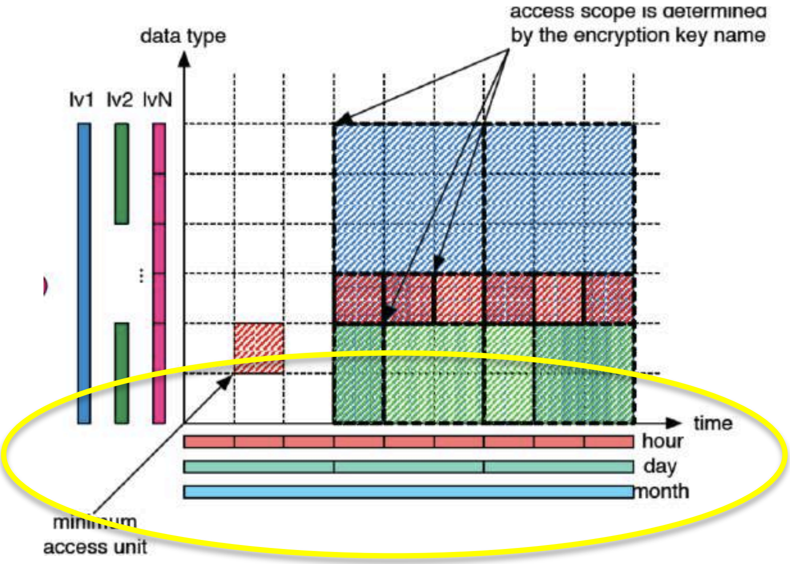


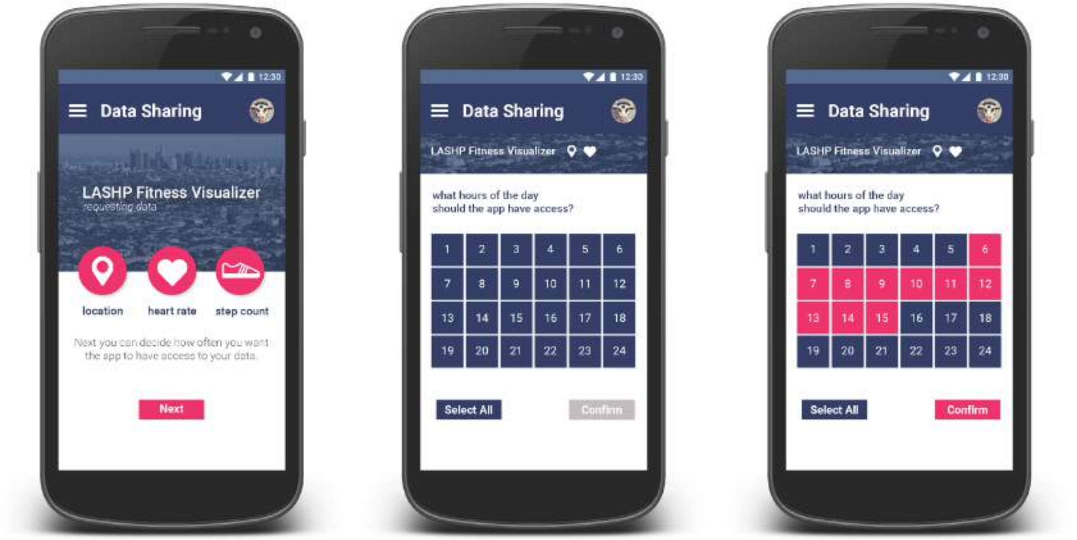
The app needs your location data to work.

Your heart rate data will help the fitness app calculate your general fitness level.

Your step count will help the app calculate the number of calories you are burning.

Authorize sharing of certain data types





Next you can decide how often you want the app to have access to your data.

what hours of the day should the app have access?

1	2	3	4	5	6
7	8	9	10	11	12
13	14	15	16	17	18
19	20	21	22	23	24

At certain times

Architectural Drivers / Challenges

- Life-long data. Enabling the user to “move” data storage from provider to provider and maintain the same namepsace.
- Usable security. Schematized trust and name-based security are conceptually simple, but work to be done on how to provide this power to developers in a simple way.
- How to best expose security choices (at selectable levels of details) to end users.
- Best type of sync protocol(s) for mobile upload, storage – processing.
- Access control between processing blocks.
- Publisher mobility.

Next

- Mobile capture application, data storage unit, identity manager, certificate assignment implemented by end of calendar year, including schematized trust and name-based access control.

- Visualization app and end-user testing. Open mHealth team, and Cornell Small Data group review after that.

- Chainable data-processing units, using Named Function Networking (design ongoing with C. Tschudin, Univ. of Basel.)

- Consideration of sync-based protocol designs for many-to-many, intermittent communication.

- Name confidentiality approach.

Climate Applications (mini-update)

Christos Papadopoulos, Chengyu Fan, Catherine Olschanowsky, Susmit Shannigrahi, Steve DiBenedetto, and others.

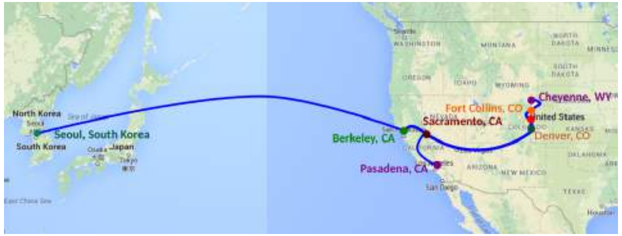
Managing Scientific Data with NDN

Distributed, synchronized catalog of names and services

- Common functionality: publishing, discovery, access control, etc.
- Search and retrieval UI
- Platform for further research and experimentation

Research questions:

- Namespace construction, distributed publishing, key management, UI design, failover, etc.
- Functional services such as subsetting
- Mapping of name-based routing to tunneling services (VPN, OSCARS, MPLS)



Science testbed

- 10G testbed (courtesy of ESnet, UCAR, and CSU Research LAN)

Nodes strategically located near scientific data (climate +HEP)

See named-data.net for publications.

Work supported by NSF #1345236 and #13410999

Enterprise Building Automation & Management

Zhehao Wang, Wentao Shang, Jiayi Meng, Adeola Bannis, Jeff Thompson, Yingi Yu, Lixia Zhang, and others.

From Enterprise to Space to Thing

Objective

Explore the Internet of Things, in the context of building automation and management, from the top-down and the bottom up. (And now the middle out.)

Platforms

- Enterprise (Siemens BMS); Smart Space (Raspberry PI); Thing (Arduino)

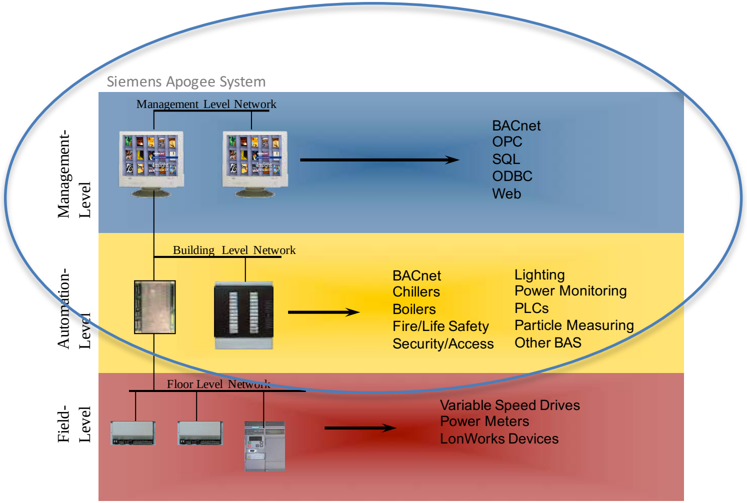
Reformulate industrial control and IoT as areas that do not rely on network isolation for security and can be fully integrated with local and global networks as necessary.

Enterprise BMS: Supporting Warehousing & Queries

Dealing with typical data query challenges became an exploration of how to decompose SQL queries, or other standard query types, onto NDN-stored data.

(Future) Data names at the NDN layer can serve the purpose of the “primary index” in SQL stores *or* log entries in more modern data stores, but need work on how to design to best support standard queries.

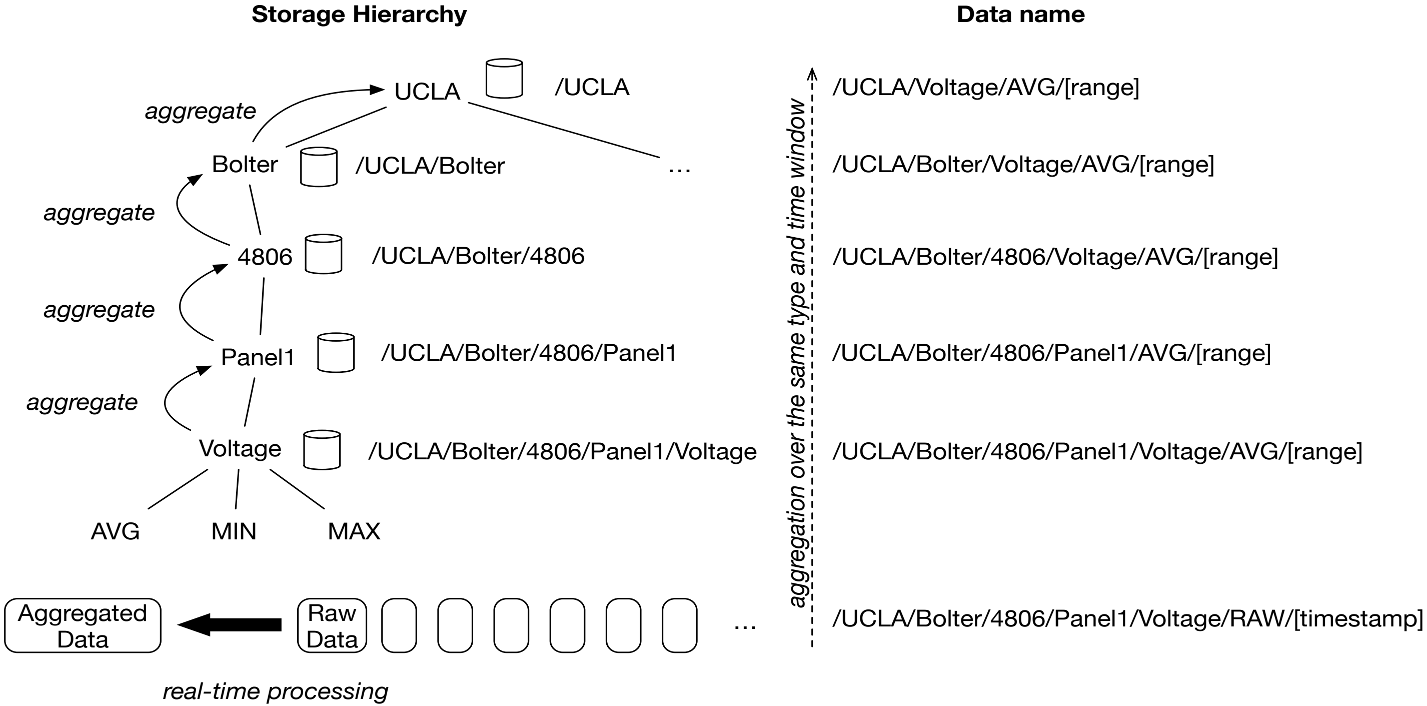
(Current) Provide hierarchical storage, keeping detailed data near the sources, and aggregating upwards. Essentially pre-selects a set of common queries and then organizes the storage to support them.



Initial work by Wentao Shang (UCLA IRL), continued by Zhehao Wang (UCLA REMAP).

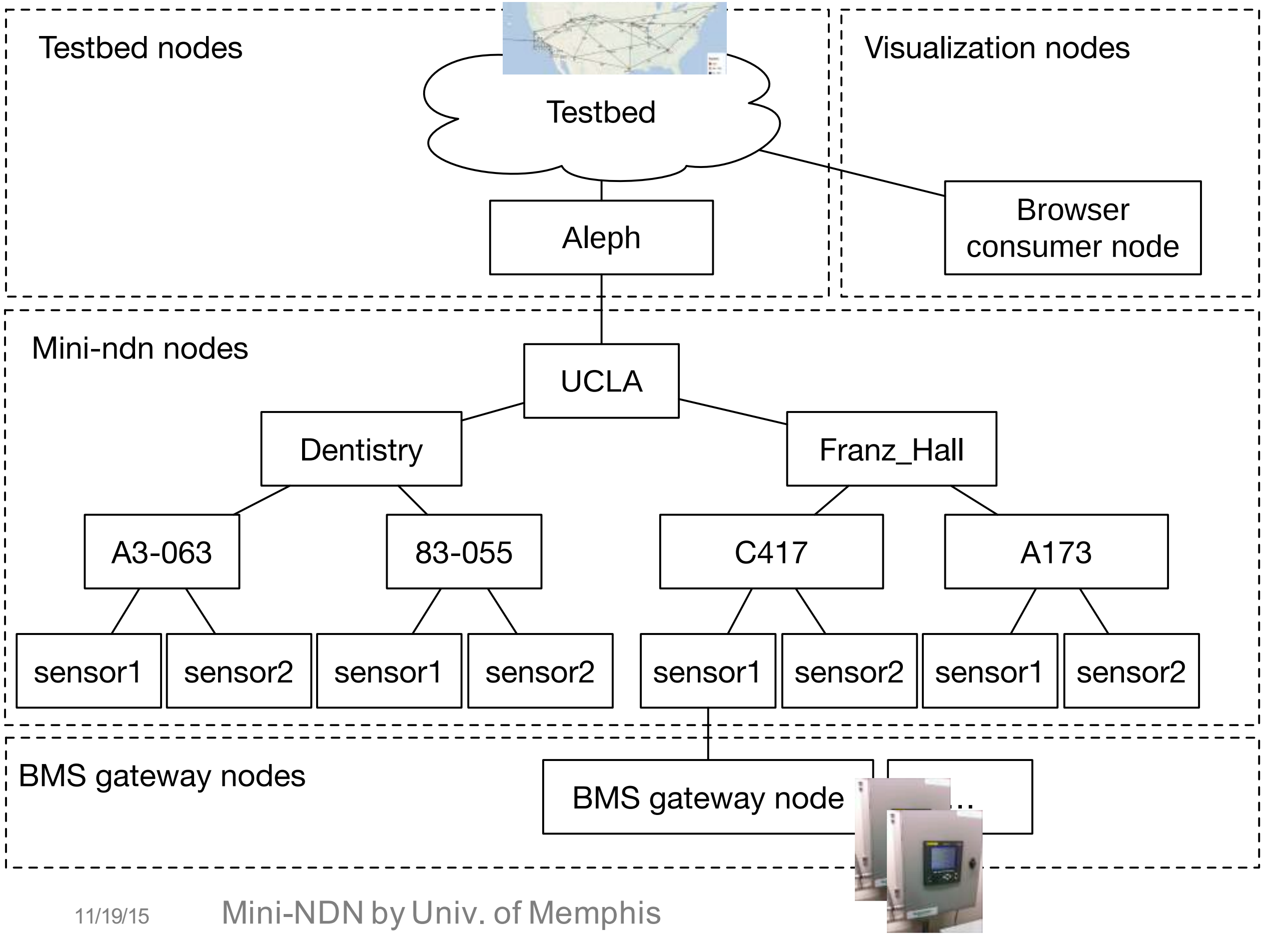
Hierarchical Storage / Schematizing Aggregation

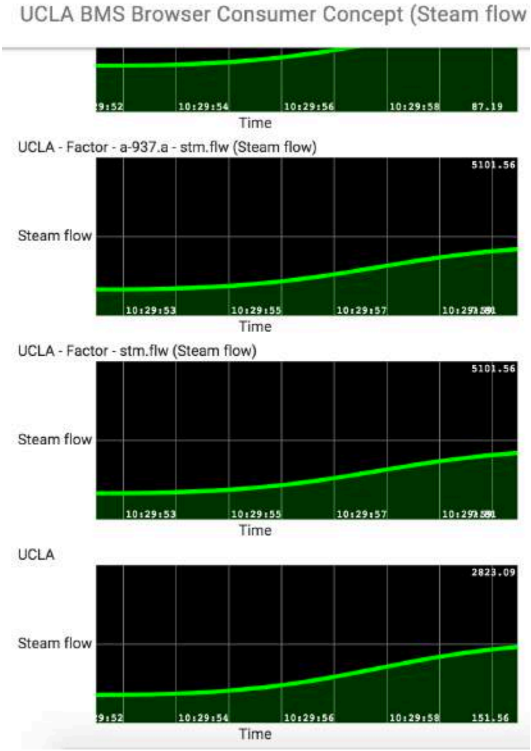
Approach: aggregate data with the same type as we move up the hierarchy



Storage Hierarchy
Data name
UCLA
/UCLA
aggregate
Bolter
/UCLA/Bolter
…
aggregate
4806
/UCLA/Bolter/4806
aggregate
Panel1
/UCLA/Bolter/4806/Panel1
aggregate
Voltage
/UCLA/Bolter/4806/Panel1/Voltage
AVG
MIN
MAX
Aggregated Data
Raw Data
…
real-time processing
aggregation over the same type and time window
/UCLA/Voltage/AVG/[range]
/UCLA/Bolter/Voltage/AVG/[range]
/UCLA/Bolter/4806/Voltage/AVG/[range]
/UCLA/Bolter/4806/Panel1/AVG/[range]
/UCLA/Bolter/4806/Panel1/Voltage/AVG/[range]
/UCLA/Bolter/4806/Panel1/Voltage/RAW/[timestamp]

Mini-EBAMS test environment




UCLA BMS Browser Consumer Concept (Steam flow

UCLA - Factor - a-937.a - stm.flw (Steam flow)

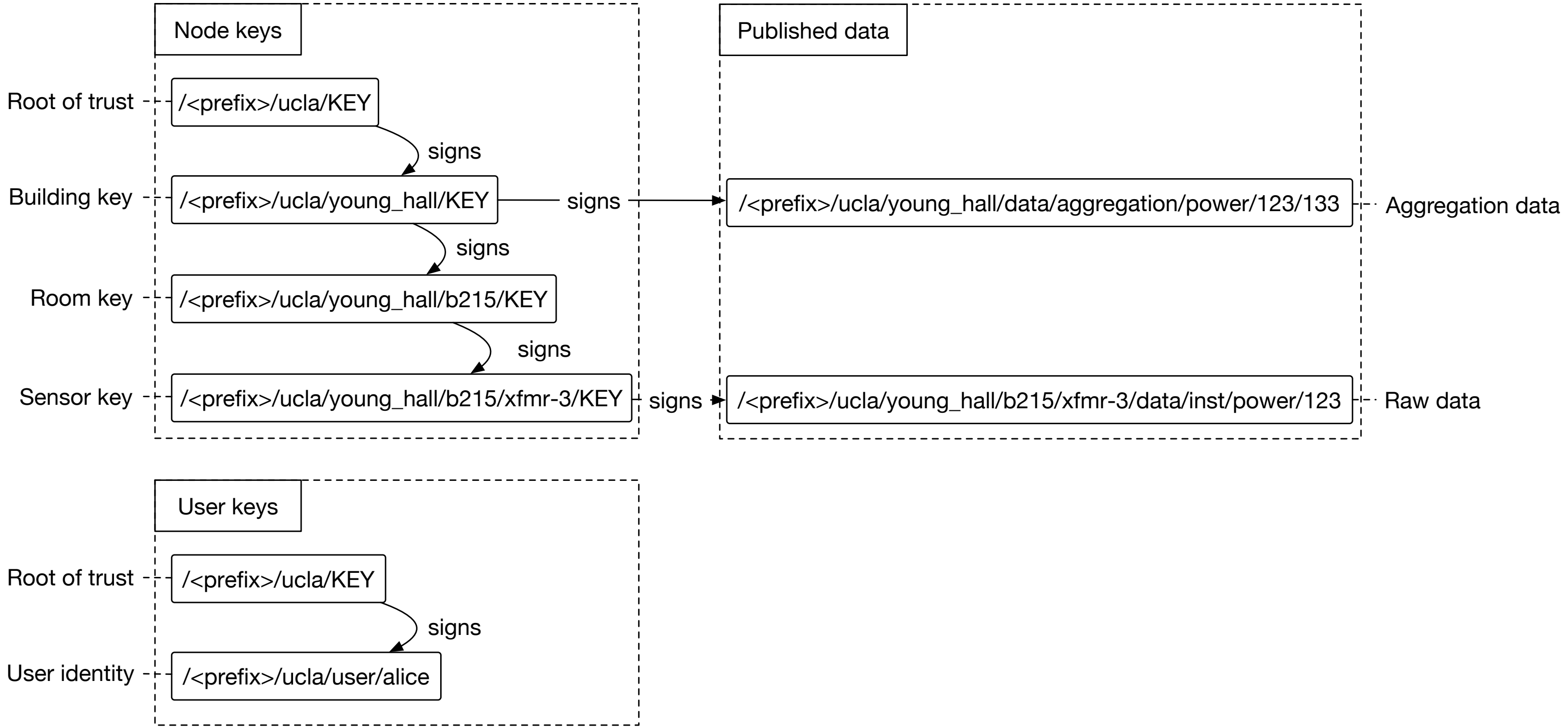
UCLA - Factor - stm.flw (Steam flow)

UCLA

Mini-NDN by Univ. of Memphis

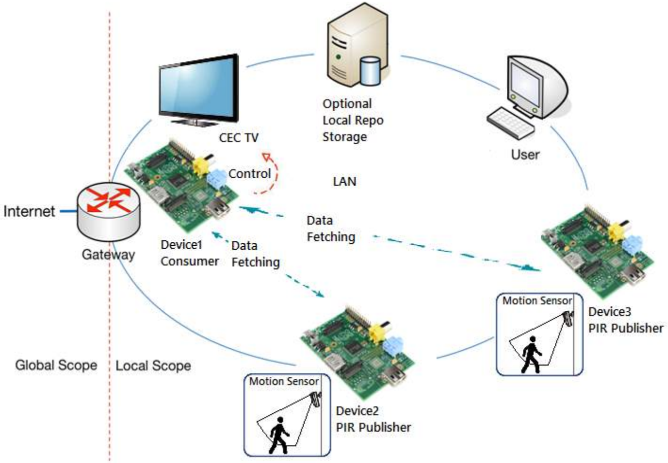
Schematized Trust Example

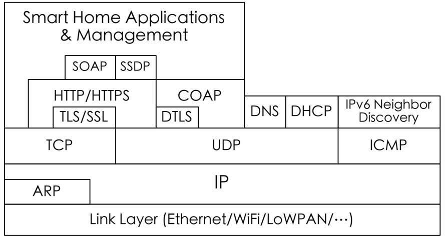
For general approach, see NDN Tech Report #30.


Node keys
Published data
Root of trust
/<prefix>/ucla/KEY
signs
Building key
/<prefix>/ucla/young_hall/KEY
signs
/<prefix>/ucla/young_hall/data/aggregation/power/123/133
Aggregation data
signs
Room key
/<prefix>/ucla/young_hall/b215/KEY
signs
Sensor key
/<prefix>/ucla/young_hall/b215/xfmr-3/KEY
signs
/<prefix>/ucla/young_hall/b215/xfmr-3/data/inst/power/123
Raw data
User keys
Root of trust
/<prefix>/ucla/KEY
signs
User identity
/<prefix>/ucla/user/alice

Drilling Down: **Smart Space (RPI)**

- UA-led work (with NeTS EAGER) continues in the smart home context:
    - **Device bootstrapping**: the initial exchange of keys between the device and the controller.
    - **Device discovery and configuration**: how does a new device learn about existing devices in the home – further work on capabilities.
    - **Access control**: which device can access which data/device, and do so efficiently.




Smart Home Applications & Management

SOAP | SSDP

HTTP/HTTPS | COAP

TLS/SSL | DTLS

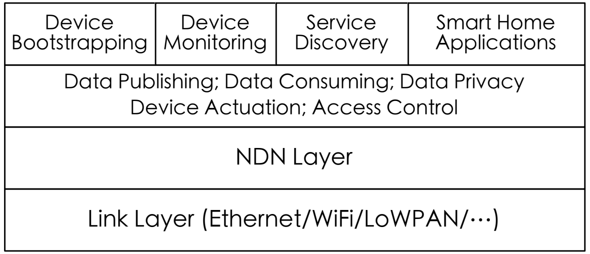
DNS | DHCP | IPv6 Neighbor Discovery

TCP | UDP | ICMP

IP

ARP

Link Layer (Ethernet/WiFi/LoWPAN/···)


Device Bootstrapping | Device Monitoring | Service Discovery | Smart Home Applications

Data Publishing; Data Consuming; Data Privacy
Device Actuation; Access Control

NDN Layer

Link Layer (Ethernet/WiFi/LoWPAN/···)

Comparison with COAP/DTLS

- NDN IoT Kit on Raspberry PI - github.com/named-data/ndn-pi
- COAP comparison - after one-time prefix registration overhead, comparable or improved communication burden.

	Lighting device		Controller		Client device	
Protocol	**Sent**	**Received**	**Sent**	**Received**	**Sent**	**Received**
CoAP-DTLS	279	350	347	369	719	626
NDN-HMAC	140	99	265	104	203	405

(values in bytes)

A. Bannis & J. Burke. "Creating A Secure, Integrated Home Network of Things with Named Data Networking," in submission. (Preliminary version available as NDN TR #35.

Drilling Further: **Constrained Devices (Things)**

Lots of industry and student interest here.

Initial work: Arduino port of interest / signed data exchanged. Next: forwarding.

NDN-CPP Lite

- Lightweight C++ lib targeting Arduino-class devices
- No assumptions about memory model
- No support library dependencies
- Application creates and supplies memory
- Shared C core with NDN-CPP:
  - Packet encoding/decoding
  - Network transport (UDP, etc.)
  - Only standard C library (strlen, math.h, etc.)

Sample Applications

- Register a prefix, receive an interest, return an HMAC signed data packet holding an analog measurement
- 28 kilobytes when compiled
- Modifications for Arduino:
  - YunClient bridge for TCP
  - Arduino-optimized SHA256/HMAC code
  - Use Arduino native random number generator
- Hackathon 2015: Update for RFduino and BTLE. Fragmentation for Bluetooth Low Energy, following NDNLPv2 with some modifications

https://github.com/named-data/ndn-cpp/tree/master/src/lite

https://github.com/jefft0/ndn-btle/

Architectural Drivers / Challenges

- Management of access control is challenging to schematize outside of enterprise or hierarchical applications.
- Also need to provide patterns and tools to manage tradeoffs between access granularity, key lifetime, and other factors in encryption-based access control.
- Lots of interesting work to do on NDN-backed databases / query support, multiple names for the same data or set-based names.
- Namespace conventions for discovery, negotiation, data access
- For “things”, need to meet the baseline expectations of IoT researchers – open loop communication, power conservation, resource constraints; early evaluation and implementations suggest this is doable.

Next

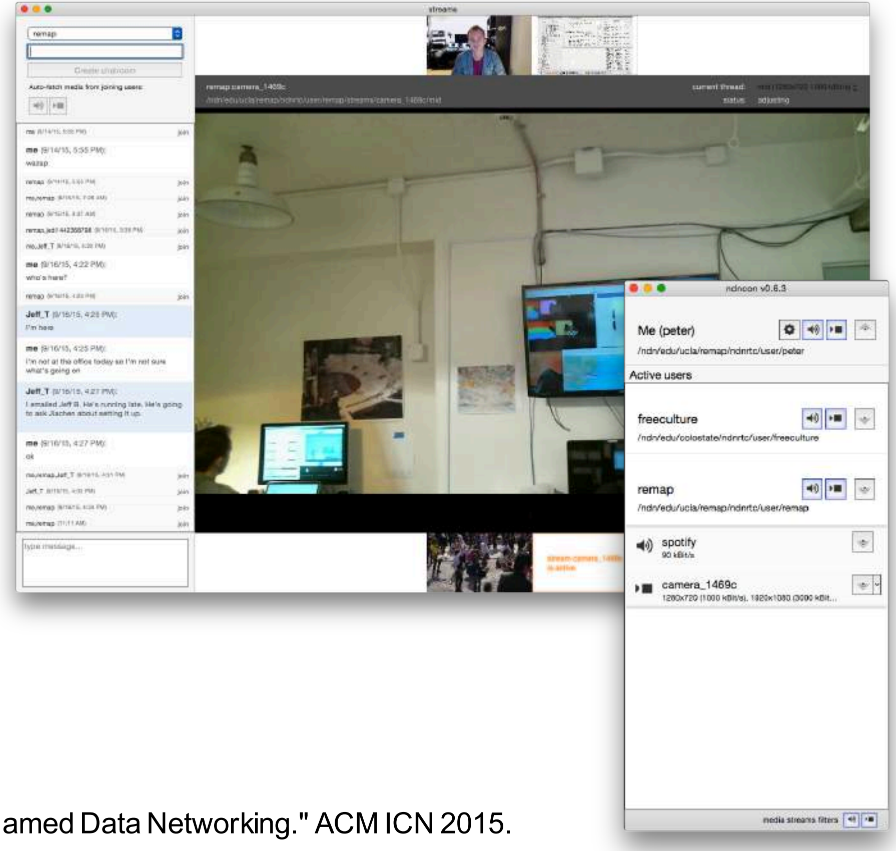
- Transitioning to system of systems work – composing end-user experiences across these platforms.
- Implement name-based access control in Mini-EBAMS.
- Apply discovery and bootstrapping ideas from smart space scale upwards (enterprise) and downwards (thing).
- Review with UCLA Facilities Management. (Towards project end.)
- Work on constrained devices, via Arduino and Bluetooth LE:
  - Finish fragmentation implementation (done for producers, finish for consumers.)
  - Create simple forwarder for constrained devices.
  - Extend NFD and/or libraries to handle lower-layer BTLE directly (currently in separate Android app). E.g., device discovery process generates name registration requests, with the Bluetooth MAC address as the prefix.
  - Specific application deployment: airbone networks, low-rate neighborhood area networks.

Mobile Multimedia Applications

Peter Gusev, Jiachen Wang, Jeff Burke, Lixia Zhang, and others.

NDN-RTC

- Third generation of live video on NDN.
- Functional videoconferencing library & app:
  - Low-latency, interactive data distribution:
    - Multi-party conferences
    - Live broadcasting
- App reformulation: RTC with no **direct communication** between peers:
  - Consumer-driven
  - More freedom for experimentation
  - Reduces signaling channel dependencies if requiredin the future
- Encourage new research using library
- Testbed traffic generation and high-load performance testing



remap

Auto-fetch media from joining users:

me (9/14/15, 5:55 PM):
wazap

me (9/16/15, 4:22 PM):
who's here?

Jeff_T (9/16/15, 4:28 PM):
I'm here

me (9/16/15, 4:25 PM):
I'm not at the office today so I'm not sure what's going on

Jeff_T (9/16/15, 4:27 PM):
I emailed Jeff B. He's running late. He's going to ask Jiachen about setting it up.

me (9/16/15, 4:27 PM):
ok

type message...

remap.camera_1469c
/ndn/edu/ucla/remap/ndnrtc/user/remap/streams/camera_1469c/mid

status: adjusting

ndncon v0.6.3

Me (peter)
/ndn/edu/ucla/remap/ndnrtc/user/peter

Active users

freeculture
/ndn/edu/colostate/ndnrtc/user/freeculture

remap
/ndn/edu/ucla/remap/ndnrtc/user/remap

spotify
90 kBit/s

camera_1469c
1280x720 (1000 kBit/s), 1920x1080 (3000 kBit...

Gusev, P. and J. Burke. "NDN-RTC: Real-Time Videoconferencing over Named Data Networking." ACM ICN 2015.

NDN-RTC Project Today

- Achieves target 350-500ms latency for our conferences; continuing reduction.
- HD-quality capable (1080p+)
- Ongoing tests ofstreaming performance
  - Up to 7.5Mbit/sec over current testbed.
  - Isolated testbed tests (bidirectional streaming)
  - NDN testbed tests with multi-hop paths (bidirectional streaming)
- ***ndncon*** GUI OS X NDN-application:
  - group text chats
  - screen sharing
- Ubuntu build support and headless app
  - Help from Luca Muscariello (Orange).
  - For upcoming scaling tests.

Apply to other low-latency data types

OpenPTrack: Open source positional tracking.

Based on the Robot Operating System (ROS)

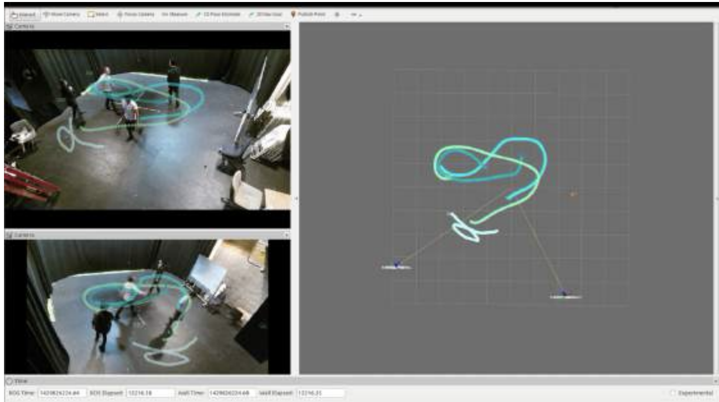
(Developed as part of NSF Cyberlearning IIS-1323767)

NDN output of 30Hz tracking.

Used in six thesis projects presented pubicly.

Leverage what we are learning from NDN-RTC to support low-latency sensing.

Future project: internal ROS messaging via NDN.



Architecture Impact

- NFD: Revised retransmissions strategy
  - App retransmission was suppressed until Interest times out in PIT
  - Varying Interest lifetime is risky when data is not produced yet or network conditions change
  - BestRoute2 strategy allows early app retransmission without giving up Interest lifetimes
- Libraries (here, NDN-CPP): Library support for app-level PIT
  - Common low-latency case: handle Interests that arrive before data is ready
  - Need to store Interests in producer-side PIT
  - Same approached used in OpenPTrack real-time person-tracking
- New ideas on consumer-side characterization of the net, generalization of RTT
- Testbed/NFD: Performance stress-tests (ongoing)
  - 3-9Mbit/sec data streams per producer
  - 9Mbit/sec: ~1000 Interest/sec, ~900 data segments/sec
  - Traffic generator for the testbed

Architecture Drivers / Challenges

- How to robustly detect arrival of latest data?
  - Still targeting no direct producer-consumer communication
  - Current approach:
    - observe intrinsic network indicators
    - cached (stale) data arrival copies Interest expression pattern
- How to efficiently encrypt media without losing NDN advantages?
  - Depends on application objectives
  - Leverage broadcast encryption and other schemes
- How to achieve inter-consumer synchronization?
  - While preserving no direct communication
  - Consider varying network conditions
- Where is historical data stored?
  - Depends on application objectives – drill down into use cases.
  - Audio/Video, chats, attachments, etc.
  - Historical data trust model

Next

- Adaptive rate control (with Panasonic Research)
- O(1000) one-to-many broadcast test (UCLA & WUSTL)
- Verification of end-user multipath support (related work at Orange/Cisco)
- Performance improvements, engineering enhancement
  - Evolve sync-based chat mechanism to multiparty signaling channel.
  - Provide efficient data-centric security.
  - Enable auto-selection of super-peers to provide mixing and coordination capabilities to large multiparty conferences.
  - Provide NDN-native scalable video coding (SVC) support.
  - Improve / replace latest-data-chasing and network characterization used for latency minimization, rate adaptation and congestion control
- Fundamentally new capability
  - Unified library and publisher protocol supporting conferencing, live streaming, playout - different consumer access patterns on the same namespace.
  - Direct SVC layer, image component access by names
- Working on wide adoption by NDN community :)

Recap

(More to come in Beichuan’s slides)

Things learned so far

- Named data approach works across a variety of application types and requirements; lots of potential for new app-level interoperability and for bridging IoT with the “traditional” Internet.
- Plus, schematizing security based on names holds promise for usable data-centric security that is both powerful and consistent; evolution driven by netenv requirements.
- Shifting protocol design from conversational to dissemination-oriented requires a new way of thinking but enables more apps to use the same data.
- Applications *will* benefit from sync and other higher-level protocols/APIs as they evolve.
- Now approaching practical usefulness as a middleware replacement. Necessary for that path is to make NDN best-in-class at autoconfig, mobility, and security features
- **Naming design remains a key challenge:**
  - Naming affects how the applications can query (efficiently) for the data. Therefore we need schemes that best fit the access semantics. To best support different types of queries, we may need multiple naming hierarchies.
  - Naming is also related to the structure of underlying storage. We eventually take the bottom-up approach and use the storage hierarchy as the main driving factor in the naming design.
  - Naming can be used to schematize trust and access control, but this places other demands on the design.