NDN Network Environment: Enterprise Building Automation and Monitoring Systems

Jeff Burke

January 15, 2015

Abstract

The abstract.

1 Introduction

As part of the NSF-supported NDN "Next Phase" research from 2014-2016, the NDN project team has selected two network environments, **Open mHealth** and **Enterprise Building Automation & Management**, and one application cluster, **Mobile Multimedia**, to drive our research, verify the architecture design, and ground evaluation of the next phase of our project. The two environments represent critical areas in the design space for next-generation Health IT and Cyberphysical Systems, respectively. They also extend work started in the previous NDN FIA project on participatory sensing and instrumented environments to focus on specific application ecosystems where we believe NDN can address fundamental challenges that are unmet by IP. Based on the successful initial results of previous NDN research, we have identified Mobile Multimedia as an application area of cross-cutting relevance, motivated not only by the network environments above but our team's desire to further develop NDN by using it for our everyday communication.

This technical report provides background information on the **Enterprise Building Automation and Management** network environment including key application challenges faced using IP and describes the design for a pilot application that the NDN team is building. It serves as the primary design document for this application.

1.1 EBAMS Background

For our purposes, Enterprise Building Automation and Management covers the intersection of three critical sub-areas: *industrial control systems* (ICS), including supervisory control and data acquisition (SCADA) and so-called smart grid [3], *enterprise networking*, and the *Internet of Things* (IOT) movement [1]. Enterprise BAS and BMS are environments that bring both critical infrastructure considerations of ICS with exciting visions for the everyday built environment of IoT. In this domain, significant engineering challenges have emerged along with the promise offered by the convergence of networking in ICS with traditional IT, a sea change described by NIST in their ICS security review [11].

BAS and BMS are software/hardware systems that perform control, monitoring and management of heating, ventilation and air conditioning (HVAC), lighting, water, physical access and other building components. Their distributed, heterogeneous nature leads to a variety of challenges. The IP protocol suite is increasingly used to network their components and as such is now a fundamental substrate of new buildings. However, IP networks suffer from limitations that impact innovation and trust in networked building systems, which we believe can be addressed with NDN.

BAS/BMS pose complementary challenges to those of mHealth. They integrate hundreds to thousands of data collection and control points often implemented by special-purpose embedded devices and managed by a single enterprise. Some devices are mobile and others are fixed; some devices are power-constrained and wireless while others are hard-wired and continuously communicating. Figure 1 shows a few components of a typical system, including thermostats used for adjusting HVAC, lighting control and energy monitoring.

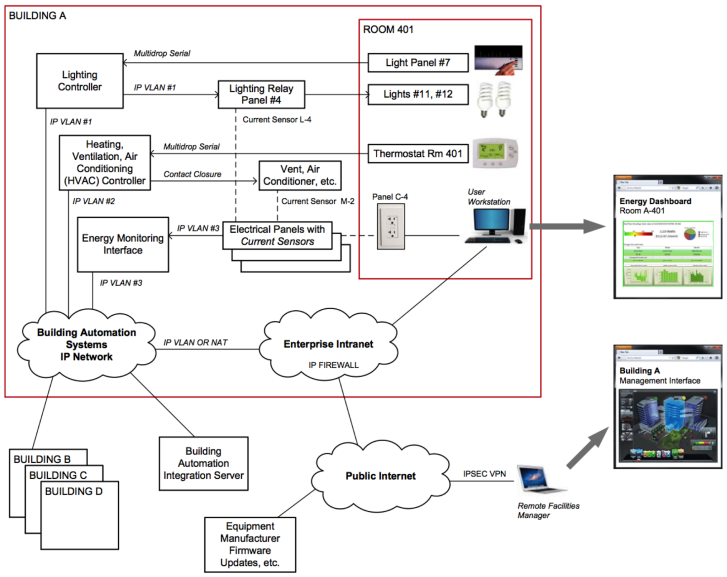

Figure 1: Example of building automation and management components, including sensors, controllers, and interfaces to both the public Internet and an enterprise intranet. Securely integrating these heterogeneous components, while simplifying application development, is an important challenge posed by this network environment.

Security is a fundamental and critical challenge [6]. Like other ICS, BAS/BMS typically employ physical or logical isolation of the network as a primary security measure, which limits interoperability and integration.[1] They also use a mixture of proprietary and open protocols, only a few of which offer intrinsic security. In many networks, there is no intrinsic security at lower layers, and only channel- and perimeter-based security above that, via SSL/TLS, VPNs, and routing configuration. Given the importance of integrating subsystems for applications such as energy management, fault detection, and synchronization, network segregation and closed protocols are not a viable long-term approach. Figure 1 shows an environment that supports access to control and monitoring from intranet web sites, but makes "air gaps" impossible and logical network separation hard to engineer and maintain for normal enterprises.

For application developers, there is a fundamental mismatch in BAS/BMS between how network applications are authored (typically data-centric) vs. fieldbus and IP network abstractions (host- and device-centric). Addressing is spread across many layers, most of which are not routable. There is a great heterogeneity in protocols, especially at the fieldbus level. Typically, application developers and platform manufacturers implement abstraction layers in middleware [8], often complex, proprietary, and challenging to configure.

In addition to the protocols themselves, significant application logic is bound up in network configuration not accessible to developers or users: 1) VLANs, IP subnetting, and routing configurations enforce boundaries between systems; 2) firewall configurations describe brittle rules for system access, which can be difficult to change; 3) keys and certificates for SSL connections and VPNs may identify connections; 4) VPN configuration and enterprise authentication hold remote network access permissions. None of these are typically visible or accessible to application software in traditional systems. In fact, they represent important system control logic that is often replicated ad-hoc in application configurations. A simple example is how an application must be configured to know that 192.168.2.1/24 is lighting and 192.168.3.1/24 is HVAC, which is site-specific and meaningless to an application. Such configurations also bring brittleness to changing topologies and devices.

Our selection of digitally-controlled cyberphysical systems as a network environment is inspired by the practical goal of enabling more efficient operation of buildings, improved comfort and control for occupants,

[1] In many cases, systems are left exposed to the Internet inadvertently. Only a few searches on `http://shodanhq.com/` make this abundantly clear.

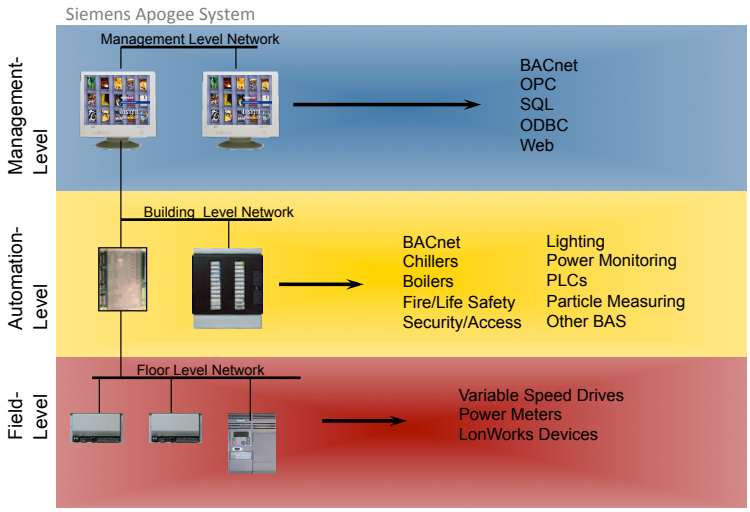

Figure 2: Levels in a Siemens Apogee system. The EBAMS network environment focuses on the "management" and "automation" levels.

and new opportunities for understanding the interactions between elements of our built environment.

1.2 Relationship to NDN-IoT

The EBAMS research has much in common with the "Internet of Things" (IoT) vision and related NDN research. For now, we consider IoT applications separately (and not in this document), motivated from consumer experience and smart home deployments. That research focuses on a bottom-up approach to networking devices using NDN, including issues such as NDN support on resource-constrained devices, bootstrapping and name assignment. The EBAMS network environment focuses on enterprise-scale networking of industrial control and monitoring technologies.

Figure 2 shows the "levels" of an industry-standard Siemens Apogee system. Our NP research focuses on the "management" and "automation" levels, while the IoT work has more in common with the "field" level .

1.3 Challenges of IP

- **Addressing spread across many layers** (e.g., VLAN, host IP, port, device number, etc.) that over-emphasizes gateways over actual sensors/actuators.
- Legacy protocols **rely on logical / physical isolation** of control and monitoring networks from IT systems. Use of IP protocol provides many new integration possibilities but new security risks; **traditional perimeter and channel-based security insufficient**.
- Many devices dont have user interfaces. **Discovery and bootstrapping** still relies on DHCP, TFTP, etc. or manual configuration.
- Middleware has been proprietary and often heavyweight. To get data-centric communication, often building on SOAP/HTTP, etc.
- COAP also proposes a request-response model but still the **channel-based security** of DTLS as the primary approach, with IPSec as an alternative. Multicast support is limited across all of these options.

1.4 Benefits of NDN

- Massive **addressing simplification**, with a potential for huge impact when scaled to the enterprise. Simpler network infrastructure needed to deploy complex monitoring and automation.

- New ways of working with edge resources that **de-emphasize gateway addressing** while preserving support for topological heterogeneity.
- Lighter-weight, **data-centric security** options easier to develop, with data verification intrinsically part of the architecture.
- **Caching and storage integration** may provide significant advantages in distributed storage at all levels of the architecture, increasing
- data availability without power increase.
- Intrinsic multicast; **many-to-many communication easier to deploy**.

1.5 Research Objectives

Naming and application design. The NDN architecture can reflect application knowledge (and needs) directly. Application logic for accessing devices can be captured in naming patterns and cryptographic signatures directly supported by the network. To support BA system integration, we aim to lower the cognitive distance between network protocol details and application requirements, so that complex BAS applications are easier to write, debug, and maintain. For example, an NDN-based BAS would use hierarchical naming rather than addresses and port numbers, removing the need for middleware to translate. We will have to develop consistent, granular, application-specific naming of data sources and control points, to support changing topologies and device configurations. NDN-network-based BAS could also make routing or forwarding decisions based on application-level semantics, essentially impossible in IP networks.

Trust and security. The trust model for this environment will emerge from the administrative organization of the enterprise and functional relationships among components. We believe these relationships can be expressed in the data and control namespaces, allowing straightforward trust verification in the applications. Extending our prior work in authenticated control, we plan to develop a system security approach that secures the data directly through cryptographic signatures on data packets and optional encryption of content. As a result, anyone equipped with the right credential (in the trust framework) can securely access, configure, and/or control devices using the same data name from any location in the enterprise. NDN-capable devices can then communicate on the network with authenticated messages, rather than relying on connection-level or physical segregation, or authentication present for user interface only.

Embedded and real-time support. To examine how NDN will work at all layers of BAS and BMS, we must consider embedded and real-time systems, many of which have recently transitioned to IP-based communication. Embedded platform support through both gateways and lightweight stacks for low-capability devices, including "hard" real-time communication when appropriate support exists at lower layers [7, 10].

1.6 Collaboration

UCLA Facilities Management has agreed to act as domain experts and help define the practical requirements of this network environment. UCLA's currently deployed building management system has approximately 150,000 points of monitoring across the campus, potentially growing to over 400,000 points in the next five years. It is the largest installation on the West Coast for Siemens building systems after Microsoft. UCLA's IT, DDC (direct digital control), and engineering staff will interact with the NDN team to enumerate their requirements, challenges, and limitations. As part of the previous EAGER award, UCLA FM has already helped us install a dedicated Siemens electrical demand monitoring system for a laboratory space for research. We expect they will also help us install a dedicated server that will provide near real-time access to 20,000 points worth of data from campus' operational systems, probably about 10 buildings worth of data. We plan to use this server as a gateway to our own NDN testbed.

Pilot application collaboration:

- UCLA REMAP
- UCLA IRL
- U. Michigan

- U. Arizona
- U. Memphis
- WUSTL

1.7 Proposed Milestones 2014-2016

- Review limitations in current IP-based architecture, for Facilities Management needs. (Y1)
- Design NDN namespace, repository, trust and communication model for use cases, such as energy management, new building commissioning, feedback control. (Y1; updated in Y2)
- Implement low-level NDN applications, such as energy management data gathering. (Y1)
- Preliminary embedded platform support. (Y2)
- Integrate live UCLA building data into the NDN testbed, mirroring data from 10-20 UCLA buildings. (Y2)
- Implement high-level NDN application for enterprise building monitoring, based on the above data, applying distributed 3D visualization work done in the first FIA project. (Y2)

2 Related Work

2.1 Suggested reading

- Stouffer, Keith, Joe Falco, and Karen Scarfone. "Guide to industrial control systems (ICS) security." NIST Special Publication (2013): 800-82 Revision 1.
- Shang, W., Ding, Q., Marianantoni, A., Burke, J., Zhang, L. Securing Building Management System Using Named Data Networking. (to be released as a TR), 2013.
- Burke, J. Gasti, P., Nanthan, N., Tsudik, G. Securing Instrumented Environments over Content-Centric Networking: the Case of Lighting Control. Proc. IEEE NOMEN, April 2013.
- Dawson-Haggerty, Stephen, et al. "BOSS: building operating system services." Proceedings of the 10th USENIX Symposium on Networked Systems Design and Implementation (NSDI). 2013.

2.2 Building Operating System Services (BOSS)

Observation: Existing buildings are not "programmable" because there are no layers of abstraction for devices. So, applications are not portable and access to building systems is not protected.

Proposal: Provide an "operating system" to "knit together existing pieces of infrastructure, Internet data feeds, and human feedback into a cohesive, extendable, and programmable system."

Support multiple, fault-tolerant applications running on top of the distributed physical resources in large commercial buildings.

Six subsystems:

1. Hardware and access abstractions;
2. Naming and semantic modeling;
3. Real-time time series processing and archiving;
4. Control transaction system;
5. Authorization;
6. Running applications.

See Figure 3 for their relationship.

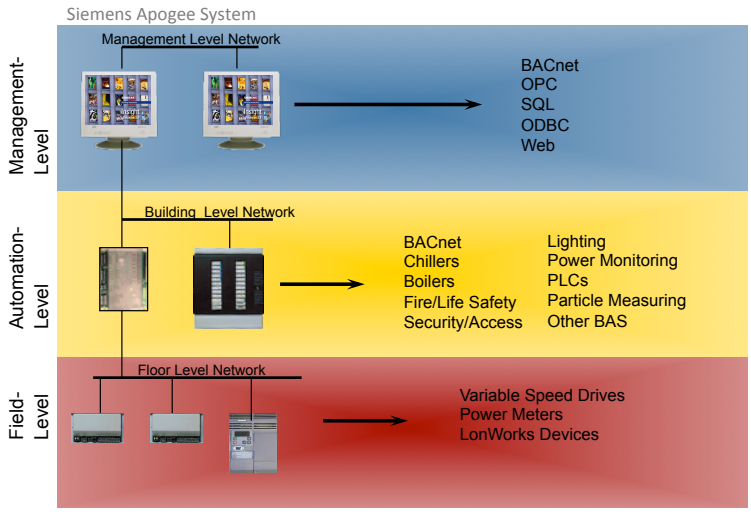

Figure 3: Architecture of Building Operating System Services (BOSS) from UCB [2].

2.3 Previous work by the NDN team

This network environment continues work that began in the original NDN project on authenticated lighting control and was extended through an EAGER in 2012-2013 to explore sensing and building management. The current and ongoing research, its results, and our evolving testbed are described in Section **??**.

3 Pilot Application: NDN-EBAMS

Our pilot applications: **NDN-EBAMS** deployment at UCLA.

Build an NDN-based collection, storage, and query system for real UCLA Facilities Management data coming from the Siemens building monitoring system. (Target 10k points at up to 1Hz in 2016.) We are initially focused on *read-only* access to sensing data.

Scope of the data: the 800 or so points of monitoring that we just got access to generate 24M rows per year and cover a few buildings and a few data types. We hope to scale to 10k pts by the end of 2016. Points are monitored no faster than 1 Hz. We get changes only, so time stamps are irregular.

3.1 Application Requirements

3.1.1 Naming and application design

- Reflect system and physical world knowledge in the naming
- Simplify application development (seeking evaluation approaches)

3.1.2 Trust and security

Fit NDN architectural mechanisms into security requirements

- Base on real administrative organization at UCLA
- Trust model? Hierarchical may work but may be in a different namespace from the data (see Wentaos work so far)
- Validating / scaling up encryption based access control approach using actual use cases.

(a) Monitored space. (b) Electrical demand monitor.

Figure 4: UCLA Melnitz Hall dedicated electirical monitoring system testbed.

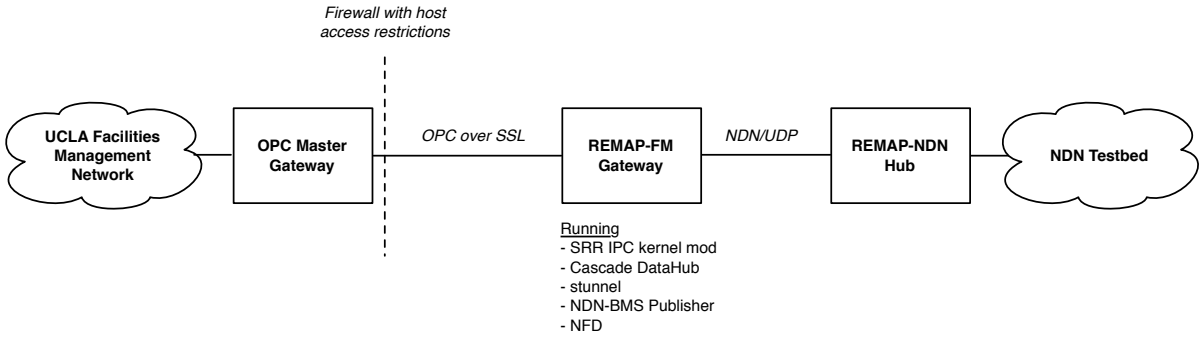

Figure 5: NDN gateway to UCLA Facilities Management building monitoring data.

3.1.3 Storage in the network

- Support the basic reporting requirements of the campus operators
- Repository design to support SQL-style queries by UCLA Facilities Management on top of distributed repositories in the style of Apache Hive.

4 Testbed

Connect UCLA Facilities Management BMS to NDN testbed with new version of Wentaos existing code. Provide NDN-JS based viewer.

4.1 REMAP Dedicated electrical monitoring testbed

4.2 UCLA Facilities Management data access

5 Design

Develop repository design and basic trust approach for offline report generation and online real-time viewing of the data as typically used by Fac Mgmt.

Challenges

- Namespace design to tackle overlapping roles of names in application data access, routing, device deployment, security, etc.
- Storage/repository design that provides familiar query interfaces.

(a) Monitored building. (b) Electrical demand monitor. (c) Chilled water flow meter.

Figure 6: Example of building monitoring at UCLA - Strathmore building monitoring.

KERCKHOFF_A264A.CHWS.RT_CV	CHW Return Temp
LIFESCI_1313.XFMR-B.DMD.INST_CV	Electricity 2 - Instant Demand
KNUDSEN_2-218.AH10.CT_CV	Cold Deck Supply Temperature
OOARH_CV	Outside Air Humidity
KNUDSEN_4-151.SF6.VFD:FREQ OUTPUT_CV	Supply Fan VFD Speed

Figure 7: Example point names from the UCLA monitoring system.

- Discovery and bootstrapping that is easier and more secure than IP.
- Trust model development and implementation (a good problem to have).
- Efficient crypto performance and group / multi-cast cryptographic challenges.
- Low-frequency, high-importance notifications (alarms).

Approach.

- A hierarchical namespace for sensor data, devices, and users that embodies intrinsic relationships in BMS applications and can be used directly for data delivery;
- Bootstrapping and ongoing management of device configuration that is simpler, more scalable and more robust than IP solutions;
- Per-packet signatures, applied immediately after data acquisition, that enable straightforward verification of data authenticity and provenance;
- Strong security through cryptographic signing to verify data authenticity and provenance;
- Encryption-based access control to sensor data, with data encrypted immediately after its acquisition, rather than relying on encrypted channels;
- Strong privacy through encryption-based access control to protect sensing data, without reliance on physical/logical network isolation.
- Identity-based authentication using NDN's (still developing) security primitives;
- Scalable user and privilege management to support enterprise-wide deployments.

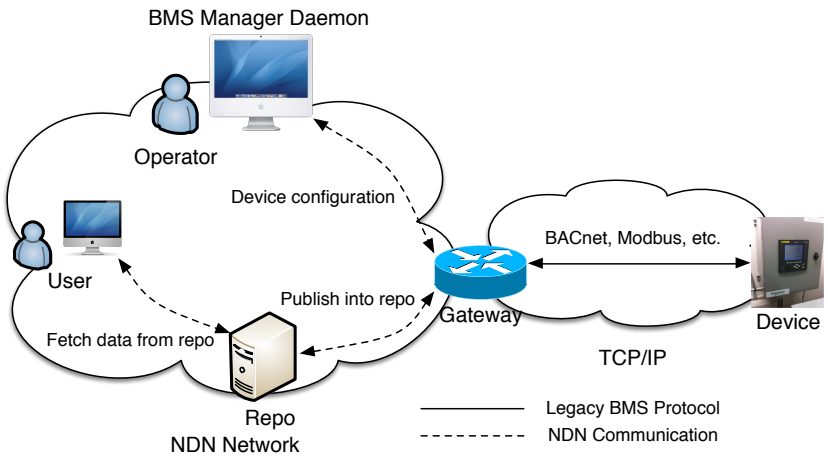

Figure 8: Major components in NDN-EBAMS, where there are potentially many users and gateways, and multiple devices per gateway.

5.1 NDN-EBAMS Application Architecture

Figure 8 shows the initial design of NDN-EBAMS with three main entities: end users, sensor gateway (including the connected devices) and a privileged manager application controlled by human operators who handle out-of-band user authentication and privilege authorization. The manager application is responsible for gateway/device and user management over the NDN network.

Gateways insert sensor data into NDN Repositories (repos), which respond to user Interests for data in the sensor namespace. This indirection decouples data generation from user fetching and provides benefits of automatic data archiving, non-interactive access control (described below), and protection from DOS/DDOS attack on gateways. In our initial implementation, a Web interface fetches the data from the repo by issuing Interest packets with proper data prefixes.

A detailed description of the initial architecture can be found at [9].

5.2 Trust and security

Only focus on read access control. No write access.

Opportunity to compare active services (e.g., broker) with passive.

5.2.1 Identity

In NDN parlance, the concept of "identity" refers to the public key owned by some entity (which could be a device, a person, or an organization). Since the keys and certificates are also published as NDN data packets on the network, each identity will receive a unique NDN name. The human-readable and hierarchical identity name can express a rich set of trust relationships and makes it easier to specify the trust policies compared to using self-certifying hashes or public key bits.

5.2.2 Trust model

The hierarchical structure in the real-world facility management department forms a natural hierarchy of trust, similar to traditional Public Key Infrastructure (PKI). In a typical deployment of EBAMS, there is a single root of trust owned by the manager of the entire enterprise. The root manager may delegate responsibilities to lower-level managers who have control over a specific department or building by signing their identities. The identity of the devices are signed by either the direct managers or other identities further delegated from the managers (such as the identity of a room or device rack).

5.2.3 Integrity

Each NDN data packet is signed by the publisher's identity. The cryptographic signature provides a strong integrity protection.

5.2.4 Confidentiality

Given the nature of EBAMS, data confidentiality is often an essential requirement. In NDN-EBAMS, confidentiality and access control is achieved through data encryption and controlling the distribution of the encryption key. The EBAMS data may be accessed by a large number of users with different types of privileges. For example, enterprise employees may have limited access to the data according to the place they work in or the department they belong to, while the utility companies may have external access to only a certain type of data (such as electricity) across the entire campus. The challenge is to design a efficient and flexible key distribution scheme that can scale to a large number of users and satisfy the diversified privilege requirements.

Should we mention the idea of APL/ACL here?

5.3 Naming

5.3.1 Data

Discover Design focuses on the actual deployment Data sources & sinks: sensors, actuators. Physical space / location will be involved in trust and should be described consistently and considered along with routing / application implications. Approach to naming/retrieving sample batches. Approach to metadata describing sensor feeds.

Hierarchical data names and simple sensor/control abstraction. See also:

Dawson-Haggerty, Stephen, et al. "Enabling green building applications." Proceedings of the 6th Workshop on Hot Topics in Embedded Networked Sensors. ACM, 2010.

Ortiz, Jorge, and David Culler. A system for managing physical data in buildings. Technical Report No. EECS-2010-128, EECS Department, University of California, Berkeley, 2010.

Potentially can use names for:

- Abstraction (allow application generalization)
- Organization (by building area, system topology)
- Metadata access (consistent approach)
- Aggregation (hierarchy inferred from names)

BOSS [2] follows these references for building metadata.

[8] Bazjanac, V., et al. HVAC component data modeling using industry foundation classes. In System Simulation in Buildings (2002).

[30] Liu, X. et al, Requirements for a formal approach to represent information exchange requirements of a self-managing framework for HVAC systems. In ICCCBE (2012).

[40] Project Haystack. http://project-haystack.org/

5.3.2 Certificates

5.4 Storage

Repository for time-series data used in BOSS [2] is readingdb, `https://github.com/stevedh/readingdb`. Their simplification of repositories is about data type more than query type: focus on time series data from points. MySQL not used because of expensive insert and poor scaling with large number of leaf keys. Low latency application interface for accessing the large repository of data at different granularity, a selection language, and a data transformation language. Enabling the construction of a pipeline of operators to the retrieved data.

Repository requirements to support sql-style access

Designing and deploying hierarchical repositories for sensor data (from sensor module to panel to building to campus scale) that provide quick access to fresh data and efficient, redundant long-term storage.

More activity on repo-ng needed?

Reasonable performance on a variety of device classes running as a component in a data producer and as a centralizer of data.

Write performance for main repo: 20,000 samples @ 1Hz on an ongoing basis.

Mechanism to export / move data offline.

Need a simple way to watch time-series prefixes.

Restore sync support for more complex namespaces.

5.5 Routing & Forwarding

Limit packets inside and outside?

5.6 Communication

Alarms Efficient support for specific types of communication patterns found in these networks, like reliable notification of rare but critical events.

5.7 User Interface

5.7.1 Website

6 Evaluation

At this early stage, we evaluate the pilot application design by discussing whether the affordances of the NDN architecture, libraries, and deployment scenarios make it easier (or possible at all) to meet the requirements of the application. To do so, we employ Green & Petre's *cognitive dimensions* framework [5] to compare a possible IP-based approach with the approach taken here. These dimensions are "descriptions of the artifact-user relationship, intended to raise the level of discourse." [4] They do not provide a comprehensive evaluation framework, merely a starting point for discussion.

Dimensions of evaluation that we will consider are:

- Abstraction gradient
- Closeness of mapping
- Consistency
- Diffuseness
- Error-proneness
- Hidden dependencies
- Premature commitment
- Progressive evaluation
- Role expressivenesss
- Secondary notation
- Viscosity
- Visibility

We also separate discussion of the prototype application from the deployed system of NDN (libraries, testbed, forwarder, etc.) and the architecture itself, following John Wroclawski's suggestion at the 2013 NSF FIA PI Meeting. [2]

6.1 Architecture

Fundamental capabilities of the NDN architecture vs. the IP architecture.

6.2 System

Codebase and testbed.

6.3 Prototype

Pilot application.

7 Open Challenges

Management challenges for detecting network faults and managing data flow in the enterprise. Student red team attacks. Encrypting / de-encrypting names at border routers for enterprises.

8 Conclusion

Acknowledgment

[2] "All hat, no answers: Some issues related to the evaluation of architecture." John Wroclawski, NSF FIA PI Meeting, March, 2013. `http://www.nets-fia.net/Meetings/Spring13/FIA-Arch-Eval-JTW.pptx`

References

[1] Luigi Atzori, Antonio Iera, and Giacomo Morabito. The internet of things: A survey. *Computer Networks*, 54(15):2787–2805, 2010.

[2] S. Dawson-Haggerty, A. Krioukov, J. Taneja, S. Karandikar, G. Fierro, N. Kitaev, and D. E. Culler. Building operating system services. *NSDI*, 13:443–458, 2013.

[3] Xi Fang, Satyajayant Misra, Guoliang Xue, and Dejun Yang. Smart grid—the new and improved power grid: A survey. *IEEE Communications Surveys & Tutorials*, 14(4), 2011.

[4] Thomas R. G. Green. Cognitive dimensions of notations. *People and computers*, V:443–460, 1989.

[5] Thomas R. G. Green and Marian Petre. Usability analysis of visual programming environments: A 'cognitive dimensions' framework. *Journal of visual languages and computing*, 7(2):131–174, 1996.

[6] Eric D Knapp. *Industrial Network Security: Securing Critical Infrastructure Networks for Smart Grid, SCADA, and Other Industrial Control Systems*. Syngress, 2011.

[7] Jork Loeser and Hermann Haertig. Low-latency hard real-time communication over switched ethernet. In *Proc. of Euromicro Conference on Real-Time Systems*, pages 13–22, 2004.

[8] T. Sauter. The three generations of field-level networks - evolution and compatibility issues. *IEEE Transactions on Industrial Electronics*, 57(11):3585–3595, 2010.

[9] Wentao Shang, Qiuhan Ding, A. Marianantoni, J. Burke, and Lixia Zhang. Securing building management systems using named data networking. *Network, IEEE*, 28(3):50–56, May 2014.

[10] Tor Skeie, Svein Johannessen, and Oyvind Holmeide. Timeliness of real-time IP communication in switched industrial Ethernet networks. *IEEE Transactions on Industrial Informatics*, 2(1):25–39, 2006.

[11] Keith Stouffer, Joe Falco, and Karen Scarone. Guide to industrial control systems (ICS) security. Technical Report 800-82, National Institute of Standards and Technology (NIST), June 2011.