KICS The Korean Institute of Communications and Information Sciences

Available online at www.sciencedirect.com

ScienceDirect

ICT Express 2 (2016) 130–134

www.elsevier.com/locate/icte

Named data networking-based smart home ☆

Syed Hassan Ahmed, Dongkyun Kim *

School of Computer Science and Engineering, Kyungpook National University, Daegu, Republic of Korea

Received 28 July 2016; accepted 14 August 2016
Available online 7 September 2016

Abstract

Named data networking (NDN) treats content/data as a "first class citizen" of the network by giving it a "name". This content "name" is used to retrieve any information, unlike in device-centric networks (i.e., the current Internet), which depend on physical IP addresses. Meanwhile, the smart home concept has been gaining attention in academia and industries; various low-cost embedded devices are considered that can sense, process, store, and communicate data autonomously. In this paper, we study NDN in the context of smart-home communications, discuss the preliminary evaluations, and describe the future challenges of applying NDN in smart-home applications.

Keywords: Named data networking; Future internet; Internet of things; Smart home

1. Introduction

Because of the recent plethora of connected devices, such as smart phones, tablets, smart watches, and laptops, the Internet industry and device manufacturers have enjoyed exponential economic growth. Similarly, Internet of Things (IoT) has gained much attention in the past few years because of its vast range of applications. According to the current definition of IoT, a collection of low-cost sensors and actuators embedded in various devices can sense, process, communicate, and react to the gathered data. In order to bring IoT into reality and benefit our daily life, various research and development efforts have been conducted by both industries and academia. For example, Internet Engineering Task Force (IETF) groups have executed several IoT projects, including IPv6 over Low Power Wireless Personal Area Networks (6LoWPAN), Constrained RESTful Environments (CoRE), Constrained Application Protocol (CoAP), and Routing Over Low power and Lossy networks (ROLL) [1]. The main objective is to support IPv6 in low-power and Lossy networks (LLNs), extend HTTP services to LLNs, and let these devices make decisions at the same time.

However, all of these efforts have one thing in common, namely, IP-based communication that forces the data to be tightly coupled with the communication channel and device-to-device addresses. In addition, the limited expressiveness of IP addressing that tends to serve both as a locator and identifier argues for a dedicated resolution system, mobility support, multicast, and enormous access under the rigorous performance requirements of IoT, hence providing challenges.

Meanwhile, the increasing demands of convergence among various heterogeneous networking devices, while keeping the information traversal robust and efficient, have motivated the research community to redesign the current Internet architecture. In this regard, named data networking (NDN) has been proposed as an extension to content-centric networks (CCNs) as a future Internet architecture [2]. NDN gives identity (i.e., a "name") to content as a "first-class citizen" within the network, in contrast to the naïve Internet, where some numeric IP addresses of the source/destination nodes and channel security are the focal points during communication. In addition, NDN secures each packet at the time of its production, enabling data caching (replication) at each node while preserving the security aspects of the data throughout the packet's lifetime.

In this paper, we take the "smart home" as a use case for IoT, as shown in Fig. 1, and propose an NDN-based smart home architecture. We also introduce a private cloud (PC) that acts as a database for storing the historical information from the home server (HS). This PC enables user(s) to retrieve the data when not in the proximity of the HS. In a smart home, we consider

* Corresponding author.
E-mail addresses: hassan@knu.ac.kr (S.H. Ahmed), dongkyun@knu.ac.kr (D. Kim).

Peer review under responsibility of The Korean Institute of Communications Information Sciences.

☆ This paper is part of a special issue entitled ICT Convergence in the Internet of Things (IoT) guest edited by Yacine Ghamri-Doudane, Yeong Min Jang, Daeyoung Kim, Hossam Hassanein and JaeSeung Song.

http://dx.doi.org/10.1016/j.icte.2016.08.007

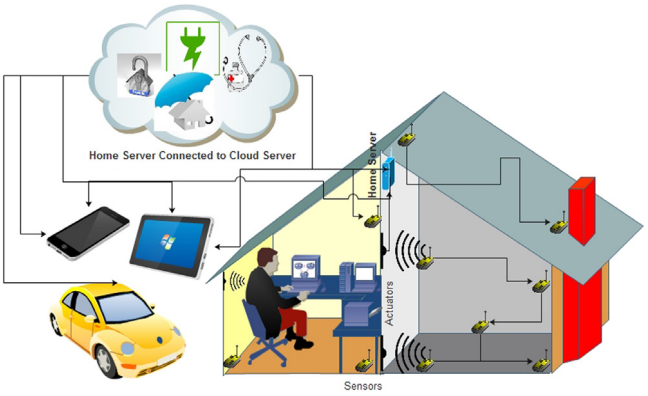

Fig. 1. NDN-based smart-home use case.

a set of sensors and actuators with a variety of applications, including energy management, security, health care, user care, and comfort. Enabling communication between those sensors via our proposed architecture is the main contribution of this work. We also verify our proposed architecture via preliminary evaluations.

2. Smart home and NDN architectures

In this section, we present an overview of a smart home from the networking and industrial perspectives, followed by NDN basics.

In 2014, Google invested in a company called Nest Labs, which was founded by iPod pioneer Tony Fedell and focused on smart-home appliances. In the Consumer Electronic Show that year (CES'14), a few interesting smart home devices were showcased, including the SleepNumber smart bed, the Belkin smart saucepan, and the Kobibree smart toothbrush. Furthermore, Cisco's CEO John Chambers mentioned at CES'14 that the IoT would spawn USD 19 trillion in IT market business [3].

Many smart-home applications involve simple operations of retrieving data and invoking some action by actuator(s). However, these two characteristics of such applications lead to complex solutions for even simple communication over IP because of the heterogeneous communication technologies and resource constraints. For example, heterogeneous IoT communication technologies such as Zigbee, Bluetooth, Wi-Fi, Wi-Fi Direct, and LAN can exist in a given smart home. Each device that provides connectivity may have various radios, wired interfaces, and serial links, all with their own addressing scheme, IP subnet, and mapping between network layer packets and application-layer messages [4]. Providing access to these devices involves an integration that can be achieved by either (a) local, application-level middleware to manage interoperability, or (b) pushing all data to cloud services, as discussed later in this paper. To provide such services, the developers and manufacturing engineers must configure and maintain mappings from interfaces to devices, and further to the intended named data and control points relating to the devices themselves. Essentially, an overlay must be created that can deal with the wide variety of underlying communication technologies, all using the TCP/IP suite of protocols, or modified versions that fit IoT resource constraints. Simply managing IP and port address assignments is not enough; in more complex scenarios, a Layer 2 configuration must be created in parallel to ensure traffic flow among heterogeneous subnets. Typically, such configuration is done outside the middleware and creates a complex set of interrelated but independently managed elements. Security requirements further complicate the picture.

Meanwhile, NDN implements a simple request/response architecture based on a state-full forwarding plane using two types of packets, called Interest and Data, both carrying URI-like names. Each node in the network acts as an NDN router and maintains three types of data structures: forwarding information base (FIB), pending interest table (PIT), and content store (CS). The FIB records matching outgoing interface(s) for particular name prefixes. To fetch content, a consumer sends an Interest packet into the network containing the name of the required content. When an NDN node receives an Interest message, it first queries matching data in its local CS. If the data are available, the matching Data packet is sent back to the consumer through the same interface. Otherwise, the node updates the PIT table with the Interest packet's name and the incoming interface. In case no existing PIT record is found, the node forwards the Interest packet over the recorded outgoing interface(s) in the FIB. When the Interest packet reaches a potential data provider or a node having a matching Data packet in its cache, a Data packet is always generated and replied back to the consumer, following the chain of the intermediate nodes. During the forwarding process, each node replicates the Data packet to all recorded incoming interfaces in the matching PIT entry, keeps a copy in the local CS, and then deletes the related PIT record. Thus, the traffic in the NDN is self-regulated, because it maintains at most one Data packet per one Interest packet in each link.

3. Named data networking in the smart home

In this section, we propose an NDN-based smart home architecture. We define a smart home as a heterogeneous place equipped with several embedded devices (for instance, sensors) capable of connecting to each other, thus supporting a variety of applications (see Fig. 2).

3.1. Home server

Similar to a traditional smart-home HS, in our work, the HS collects all the data from various sensors installed in the different areas (e.g., rooms, storage area, and back yard). For maximum availability and distinctive usage, we assume that the cached data/feedback can be stored in a PC. In most existing device-centric smart home communication designs, there is one HS that interacts with all the sensing devices for the required data and tends to send feedback to the user using Bluetooth, WLAN (Internet), and display units installed inside the home. In case of any communication error between any

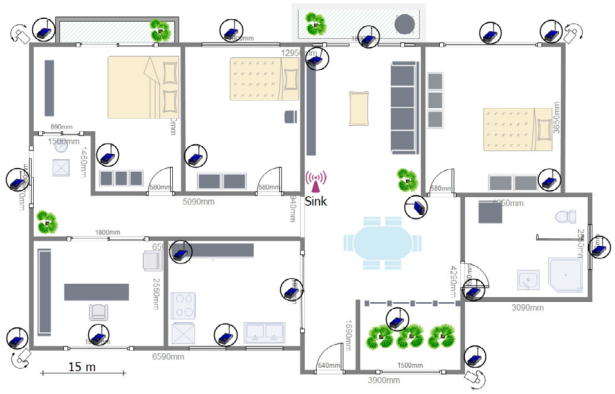

Fig. 2. Performance test: simulation setup overview.

sensor/actuator and the HS, malicious application behavior is expected. Hence, it is not ambitious to state that to date, the HS is the backbone of the smart home communication system. By applying NDN architecture, we decouple the HS and sensors/actuators, where the user can retrieve feedback directly from a specific sensor or area, by broadcasting an Interest packet for particular information, given that the user is in close proximity (i.e., inside the home) of the sensor(s)/actuator(s). In addition, we also have collected data cached and available at the HS and PC.

3.2. NDN and sensors/actuator

NDN offers viable solutions for sensor networks. In fact, it matches most of the use cases and applications that can be developed on top of sensing modules and can cope well with the potential constraints. For instance, the hierarchical naming schemes facilitate the data search and retrieval process with improved data aggregation [5]. Sensor networks are the primary beneficiaries of broadcasting any data from multiple sensor nodes towards any sink or cluster head (e.g., HS in our case). By caching each Data packet at the local CS, NDN alleviates the compromised quality of service (QoS), especially when the number of sensor nodes increases. In our architecture, sensors can communicate with the HS and with the user at the same time. In the former case, the HS broadcasts Interest packets periodically to obtain the sensed information and display it on an attached screen. In the latter case, the user can broadcast the same Interest packet and collect the data from the nearest available source, rather than piggybacking the IP address of the HS and the packet traversing from one part of the home all the way to reach the HS or PC. Thus, NDN makes it easier to provide local and global access for home automation.

3.3. Private cloud

Because there is no existing practical work on integrating cloud and NDN, we propose only an architectural overview that shows how a PC server can be beneficial to an NDN-based smart home. In our case, an application running over a PC with limited access can retrieve data from the HS through Interest packets. When the user is outside the home, the PC application can send critical data to the user via push-based forwarding in NDN [6]. Through the PC data collection, we can calculate statistics, predict critical events, alert the user, and let the HS make decisions for actuators even in the absence of the user. These decisions may include actions such as turning on/off air conditioning, lights, and automatic door locks.

3.4. Naming

As mentioned previously, data is retrieved using names. In this context, an NDN naming scheme for IoT devices (e.g., sensors in this work) in a smart home must be highly expressive and customizable at the same time. Human-readable and application-specific names support both sensor- and actuator-based tasks. The most widely used naming schemes are hierarchical and flat naming. Flat names are typically obtained through hash algorithms applied to (already existing) contents and are not easily assigned to dynamic sensed data contents that are not yet sensed/published. Therefore, we consider hierarchical naming in this work. Our naming scheme must be able to identify either tasks to perform by actuator(s) or the type of information required from the sensor(s). Thus, we identify two namespaces that can be used by the HS for these purposes. For information retrieval, an example name is `/info/humi/room1/...`, where `/info` indicates that the sensed data or humidity measurement is required from the room possessing ID 1. Similarly, HS can also retrieve the humidity level from all the deployed sensors by sending an Interest packet with the name `/info/humi/all/...`, where `/all` will collect the desired data from all the sensors instead of targeting a specific location. For a variety of information, there can be several names; for example, temperature can be named as `/info/temp/all/`. Upon receiving this information, the HS uploads this information to the PC, and if requested by the user, can forward it to the user. Because NDN-based data retrieval depends on names, and regardless of interface, the devices can cache the data. When the user needs any information related to the smart home applications on his/her smart device, the Interest packet can be generated and satisfied by the user's cell, tablet device, or nearest available resource (i.e., HS or sensors).

On the other hand, for actuators, we identify `/act` as a namespace, where a user or an application makes a decision and lets actuators perform tasks to keep the home comfortable and energy efficient at the same time. For example, if Room 1 is empty for a certain amount of time, the lightning and air conditioning can be turned off autonomously. However, such decisions require sophisticated protocols, and are beyond the scope of this work. One example name for Interest can be "`/act/room1/aircon/off/...`". To be precise, through the hierarchy of name components, a simple versioning system can be deployed to manage such cases where a producer constantly updates the content value, like the temperature in a room. This would help to manage the freshness requirement of some applications. By sharing a common name prefix for multiple content/services, hierarchical names scale better than flat names, because they facilitate the definition of name aggregation rules in the FIB, which is critical for big data. Meanwhile, this also implies that IoT applications operating in the same domain and handling information/services with global

scopes should be designed by developers with common (shared) name-prefixes.

3.5. Pull and push support

Commonly, in the NDN-based smart home it is expected that events may be triggered and information sent regardless of Interest packet arrival. Meanwhile, the current version of NDN supports pull-based retrieval with less focus on push-based communication. We classify smart-home communication into two service models: naïve pull and pushing critical data. Our proposed architecture supports both models, where HS can retrieve sensed data by broadcasting an Interest packet in a pull-based model. At the same time, sensors can also push data by sending a beacon message towards the consumer, i.e., the HS in our case, indicating the critical data arrival in the next packet. Upon receiving that beacon message, the HS can create a virtual PIT entry so that the incoming critical data is not treated as unsolicited data and thus avoids dropping it.

4. Performance evaluations

In this section, we present preliminary simulation results for our proposed NDN architecture using NS 2.34. For instance, we deployed 20 nodes in a heterogeneous manner within the home designed as shown in Fig. 2. The simulation area is set to 200 m × 200 m, and other parameters are identical to IEEE 802.11g with a simulation time of 100 s. We created three data structures for the upper layers, including the CS, PIT, and FIB (NDN basics). For evaluation, we varied the transmission range of the sensors from 60 to 140 m with a frequency of 2.4 GHz. One sink node is placed in the middle of the home that performs the role of the HS. The rest of the physical and MAC parameters are fixed according to [7]. The results shown in this paper are averaged from 20 simulation runs with a confidence interval of 31%. For comparison, we use following two metrics:

- Interest satisfaction rate (ISR):
  ISR is defined as the difference between the number of Interest packets generated and the number of Data packets received. In other words, it is the ratio of satisfied Interest packets.
- Accumulative data retrieval delay:
  This metric is calculated to find the overall delay starting from the Interest packet generation until Data packet arrival, and can be referred to as Interest-Data round-trip delay.

First, we compared the ISR against varying transmission ranges of the sensor nodes. We also tested the performance for three different intervals between Interest packets. The transmission interval of an Interest packet plays a vital role in terms of network performance, as is evident from Fig. 3. Here it is worth mentioning that we have various applications and types of sensors with different recommended transmission intervals. For example, the intruder detection sensor should have a shorter interval compared to the temperature sensor. Further, we note that the transmission range (i.e., type of sensor) has no such potential influence on the ISR, and thus we have

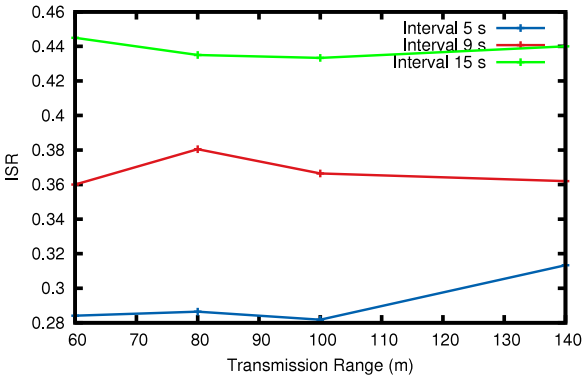

Fig. 3. ISR evaluations for various time intervals.

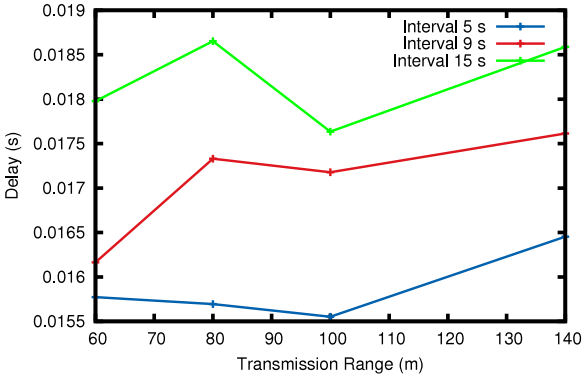

Fig. 4. Interest-Data round-trip delay vs. time intervals.

stable performance. For instance, a 15-s time interval between the generation of two Interest packets exhibits 90% better output compared to 9-s and 5-s intervals. Hence, we need to propose algorithms to optimize the Interest packet generation time interval.

Next, we evaluated the Interest-Data round-trip delay for varying transmission ranges and Interest packet intervals. It is worth noting that the ISR is especially high when the inter-Interest packet interval is high, because the probability of an Interest or Data packet collision is quite low. Hence, the gateway can successfully pull the data from nodes that are multiple hops away. Because of the successful reception of data from all multi-hop WSN nodes, the Interest-Data round-trip delay is larger because of the packet processing and route computation at each hop. This phenomenon is clearly evident in Fig. 4 for an Interest packet interval of 15 s. Hence, we conclude that the higher the Interest packet interval, the higher the ISR and delay and the lower the information freshness.

5. Conclusion

Smart-home devices have been available and the communication designs for such devices are under consideration. Currently, IoT is considered a feasible solution because of its support for heterogeneous devices. However, the IP-based IoT still requires complex algorithms for mapping the IP addresses of the devices. In this paper, therefore, we have proposed an NDN-based smart-home architecture, where sensors/actuators

can communicate on the basis of named data. In addition, we evaluated the basic performance using simulations.

Acknowledgments

This research was supported by the Ministry of Science, ICT and Future Planning (MSIP), Korea, under the Information Technology Research Center (ITRC) (IITP-2016-H8601-16-1002) supervised by the Institute for Information and Communications Technology Promotion (IITP).

References

[1] A. Betzler, et al., CoAP congestion control for the Internet of Things, IEEE Commun. Mag. 54 (7) (2016) 154–160.

[2] W. Shang, et al., Named data networking of things (invited paper), in: Proc. of First IEEE International Conference on Internet-of-Things Design and Implementation, IoTDI, Berlin, 2016, pp. 117–128.

[3] K. Xu, et al., Toward software defined smart home, IEEE Commun. Mag. 54 (5) (2016) 116–122.

[4] M. Amadeo, et al., Information-centric networking for the Internet of Things: challenges and opportunities, IEEE Netw. 30 (2) (2016) 92–100.

[5] M.F. Majeed, et al., PDF: push-based data forwarding in vehicular NDN, in: Proc. of the 14th Annual International Conference on Mobile Systems, Applications, and Services Companion, ACM MobiSys'16, Singapore, 2016, p. 54.

[6] M. Amadeo, et al., Information centric networking in IoT scenarios: The case of a smart home, in: Proc. of IEEE International Conference on Communications, ICC, London, 2015, pp. 648–653.

[7] RN-171 802.11 b/g Wireless LAN Module, Rowing Networks, datasheet available online at: http://www.microchip.com/.