A Note on Routing Scalability in Named Data Networking

Yu Zhang Zhongda Xia
Harbin Institute of Technology
Harbin, China
{yuzhang, xiazhongda}@hit.edu.cn

Alexander Afanasyev
Florida International University
Miami, USA
aa@cs.fiu.edu

Lixia Zhang
University of California, Los Angeles
Los Angeles, USA
lixia@cs.ucla.edu

***Abstract*—The Information-Centric Networking (ICN) vision shifts network communication paradigm from node-centric IP to data-centric Named Data Networking (NDN). This position paper aims to identify the impact of the above architectural transformation on network routing and forwarding: what changes, and what does not? We first clarify a few basic terminologies in networking, including addresses, locators, and locations. We then explain that, as far as the network *routing scalability* is concerned, the basic solution approaches to scaling IP network routing are equally applicable to NDN networks. We further describe several unique NDN forwarding scalability solutions, which are enabled by naming data and NDN's stateful forwarding plane, and argue that rendezvousing on data opens a new dimension of solution space in network scalability.**

***Index Terms*—Naming, Forwarding, IP, NDN, ICN**

I. Introduction

Two core questions in the design of a network architecture are naming—*what kind of identifier(s) a packet carries*—and forwarding—*how a packet is forwarded through the network based on the chosen identifier(s)*. The answers to the above questions are well-understood for today's IP networks, where packets carry source and destination IP addresses, and routers run routing protocols to set up forwarding tables, so that packets are forwarded to their destination nodes in a stateless manner. Information-Centric Networking (ICN), on the other hand, has fundamentally different answers to the above questions. ICN adopts a data-centric communication model which focuses on "what" instead of "where". As exemplified by the Named Data Networking (NDN) design [1], where each packet carries a data name, and communication is accomplished via retrieving desired data by application-layer names.

Fundamentally different from IP's point-to-point packet delivery based on addresses, in an NDN network each piece of data is uniquely named and signed by its producer, i.e., the *data producer*, and encapsulated in a network layer *Data* packet. A *data consumer* sends out an *Interest* packet which carries the name of the desired data to retrieve it. In forwarding an Interest packet, each forwarder has the freedom to choose among a variety of ways, e.g., using a forwarding table, forwarding to a single or multiple next hops, or random walk or flooding, to move the Interest towards where the requested data may reside. Each forwarder keeps track of from which interface an Interest comes in, so that the corresponding Data packet can be forwarded back to the original requester. In other words, NDN employs a *stateful* forwarding plane which keeps traces of the paths taken by Interest packets, enabling Data packets to follow along these "breadcrumb trails" to get back to requesting consumers in a hop-by-hop fashion. As far as naming and forwarding are concerned, the most significant differences between IP and NDN are 1) naming endpoint vs. naming data, and 2) stateless forwarding plane vs. stateful forwarding plane. These fundamental differences call for evolved understanding of fundamental principles.

The motivation of this paper is twofold. First, due to the fundamental difference in what gets named in IP and ICN, confusions often arise on the basic concepts of naming and forwarding in recent literature on ICN, and urgently deserve a clarification. Second, naming and forwarding are fundamental factors in addressing network routing scalability, multihoming and mobility support, which have all been long-lived challenges in networking. Addressing these challenges for ICN networking calls for a re-examination of the solution space in ICN's data-centric context.

The main contribution of this paper is a clear identification of network architectures' impact on the solution space for routing and forwarding scalability: moving from IP's node-centric data delivery to ICN's data-centric retrieval, what changes and what does not—*what solution approaches are architecture-dependent or independent*. First, we clarify the basic concepts of address/locator/location, explaining why they do not change in terms of network routing and forwarding scalability issues when a network moves from IP to NDN. Second, we explain why the basic approaches to routing scalability in IP networks are applicable to NDN routing. We then clarify what does change in the solution space, and present several existing solutions to NDN forwarding scalability; these solutions are enabled by naming data and NDN's stateful forwarding plane. We also illustrate how rendezvousing on named data opens a new dimension in the solution space.

II. Name, Address, Identifier and Locator

In this section, we first give a brief comparison between IP and NDN. Then we clarify the basic concepts of

This work is partially supported by the National Key Research and Development Program of China (2016YFB0801303), and US National Science Foundation under awards CNS-1629922 and CNS-1719403.

name, address, identifier and locator, which are architecture-independent; in particular we point out that identifiers become locators when they are announced into a network's routing plane, because these announcements establish the reachability to the identified entities, either nodes or data.

A. Network Layer Naming

In both node-centric and data-centric networking, naming provides identification (of nodes in IP, and data objects in NDN), while forwarding concerns the reachability.

Let us consider a typical use case in the Internet: a browser is to fetch a web page W from a website. As illustrated in Fig. 1, performing this task in an IP network involves at least the following three processes: 1) Name resolution: because the web page W is identified by a URL which contains a name while the network can only deliver packets to a given IP address, the Domain Name System (DNS) is used to bridge this gap in application and network layer naming, and the browser sends a DNS query to resolve W's name to an IP address W_A. 2) Routing: an IP prefix covering W_A must be announced into the routing plane which sets up an entry {IP prefix→outgoing interface} in every router's Forwarding Information Bases (FIBs). 3) Forwarding: packets carrying IP address W_A can then be forwarded *to* the website according to the established FIBs.

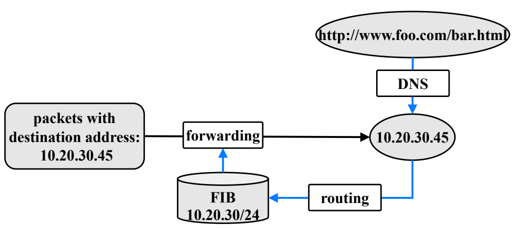


Fig. 1. A web-browsing example in IP

Fig. 2 illustrates how the same web-browsing task is carried out in an NDN network. The name in the web page W's URL is directly used for network layer routing and forwarding [1], hence no name-to-address resolution process is needed. The other two processes can be done in a way similar to those in IP: a name prefix covering the data name is announced into the routing plane which sets an entry {name prefix→outgoing interface} in router FIBs; an Interest packet carrying the data name as a request for the webpage can be forwarded *towards* the website according to the established FIBs.[2]

Node-centric network naming using addresses may only identify a device or an interface at network layer; contents and services have to be named at application layer. Consequently a namespace mapping system, such as DNS, is indispensable to

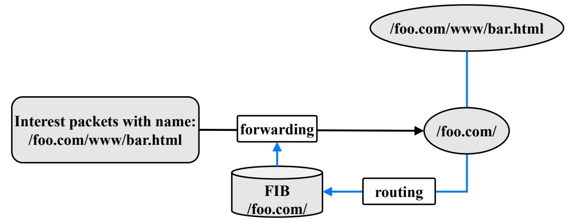


Fig. 2. A web-browsing example in NDN

bridging the gap between different namespaces. In contrast, an NDN network simply uses application layer namespace at network layer. Depending on the applications' needs, an NDN name may be a URL-like structured string or a flat bit/byte/numeric string. NDN removes IP's restriction that network layer identifiers can only name communication end-points. As far as networking is concerned, NDN names can name anything–endpoints, interfaces, commands to turn on some lights, and processing functions, in addition to data chunks from a movie or book.

B. Clarification on Concepts

From the perspective of routing algorithms and, IP address prefixes and data name prefixes have no fundamental difference in the sense that both identify forwarding directions on a graph-theoretic abstraction of network topology; an implementation difference is how long the prefixes may be. Once a prefix, either an IP address prefix or NDN name prefix, is injected into FIBs by a routing protocol, forwarders can use established FIBs to match addresses/names carried in the packets to be forwarded. In fact, a well-know link-state routing protocol, IS-IS, is designed as a network-layer-independent protocol [2].

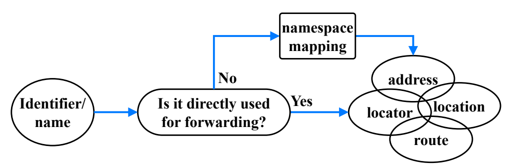


Fig. 3. Name/address-related concepts at network layer

Given the above explanation, we make a clarification on some interrelated concepts: identifier, name, address, locator, location and route. The relationships of those concepts are shown in Fig. 3. All the above mentioned terms may be viewed as identifiers, in a sense that they identify something. If an identifier/name is directly used for forwarding—set in FIBs—then it also represents a locator/address/location for what it refers to. In IP, an IP address is both an identifier of an interface and a locator in FIBs. In NDN, a data name is also a locator for the data if its name prefix is in FIBs. There is no difference between name and address in this regard, other than their variable versus fixed lengths. In fact, nowadays IP address blocks can be traded worldwide as any other types of resources, an IP address does not identify specific fixed location in general, either physical or topological. Hence we clarify that:

[1]The task of learning the names of desired data belongs to application layer, not network layer. In the web browsing example, the name of a webpage can be obtained from previous knowledge (e.g. cnn.com), or search engines, or by following links on other webpages.

[2]An Interest may hit a cache with the requested data before reaching the website, therefore we used the word *towards* here, instead of the word *to* in IP's case where the browser request is necessarily delivered to the website server.

An IP address is simply a constrained name.

In practice, however, not all identifiers are directly used for forwarding. The main reason is practical concerns regarding routing scalability. The global Internet would face scalability challenges when too many identifiers get injected into the routing system, or the reachability to identifiers exhibits too high dynamics. Routing scalability has been considered one of the few main research challenges since day one of networking, and the demands on multihoming and mobility support in recent make the problem more difficult, as the former leads to prefix disaggregation and the latter higher dynamics. A basic approach to routing scalability is introducing a mapping system which can map a non-routable identifier to a routable one, or map a high mobile prefix to a stable one, turning the prefix dynamics to frequent mapping system updates but without disturbing the overall routing system. We will further discuss this topic in the next section.

With the above understanding, we proceed to discuss the term *location-independence*. In [3], NDN is labeled as being "location-independent", "enabling communication using fixed names without concern for changing network locations." We do not believe that any architecture can be labeled either location-independent or -dependent; location-dependency, or lack of it, is determined by the dynamics of the named entities and their relation to the routing system. First, a data name can be location-dependent when this piece of data does not move, and its name prefix is announced into network routing to set up an entry in FIBs; on the other hand, a mobile in an IP network can be location-independent if it injects its address into the routing system, as Connexion did [4]. Furthermore, we stress that an identifier should not change as long as it identifies the same thing. In NDN, the same piece of data with the same name can be moved to, and retrieved from, different locations. The technical issue in hand is not so much of location-(in)dependency, but rather how the reachability to the data can be established in a scalable way.

III. IP ROUTING SCALABILITY ISSUES AND SOLUTIONS

The previous section clarifies that an identifier, be it an IP address or a data name, is a locator if it is directly announced into network routing. As far as routing announcements are concerned, IP and NDN networks have shared concerns on routing scalability. In this section, we briefly review previous work on IP routing scalability, and argue that the basic solutions to IP are equally applicable to NDN.

A 2007 IAB workshop report [5] summarized the findings from researches on network routing scalability. The size of routing tables at DFZ has been constantly growing at super-linear rate, threatening to overwhelm the growth of hardware capabilities forecasted by the Moore's Law. The report identified three major contributing factors to the growth of DFZ routing table size: no renumbering after topological connectivities change, multihoming, and traffic engineering.

Regarding the guiding principle for making routing scalable, Yakov Rekhter made an eloquent remark, also known as "Rekhter's Law" [5]:

Addressing can follow topology or topology can follow addressing. Choose one.

Since the global Internet topology is dictated by both technologies and economic relations among Internet service providers, it is infeasible to make topology follow addressing, yet all the three identified factors in the report [5] are the results of addressing not following topology, creating a dilemma. Given that Rekhter's Law is architecture-independent, one can see that both IP and NDN can face the same dilemma in routing scalability.

Two lines of work were proposed to address this dilemma in IP networks. The first one [6]–[8], commonly referred to as locator/identifier separation, strives to provide a location-independent identifier for each host, which is believed to allow IP address allocations re-organized to follow the topology. A mapping system is needed to link location-independent host identifiers to location-specific addresses.

The second line of work [9]–[12] tried to separate provider and customer address space. By assigning customer networks unique identifiers and mapping those identifiers to the connected provider networks, the transit core is separated from edge networks, removing the need for renumbering when customers change providers and facilitating multihoming.

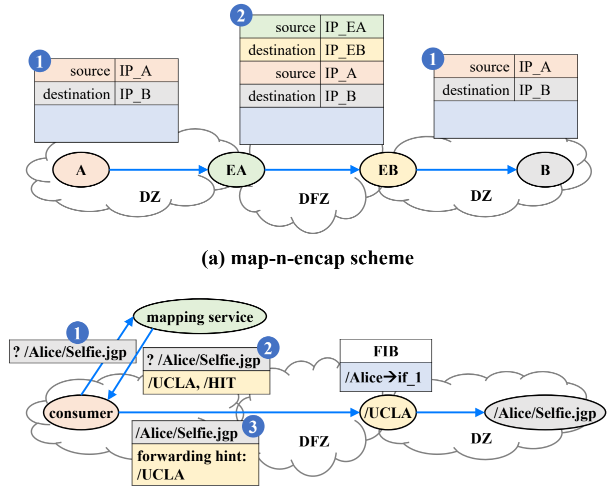
As concluded in [13], the biggest difference between the two approaches is that locator/ID separation imposes major architectural change without strong incentives, while address space separation matches the natural need for evolution due to the fact that user networks, serving completely different purposes from that of transit networks, have grown too big and powerful to bend to the addressing restriction set by transit providers.

Given two separated namespaces, either locator/ID or transit/user address, it can be observed that all routing scalability solutions attempted to control the routing table size by mapping non-routable names (those that do not get injected into routing announcements) to routable names (those that are announced to the routing plane).

We believe the above results are also applicable to NDN. To scale routing, one simply needs to make a decision on whether a given name (prefix) should be announced into the routing system, and if not, how to map this non-routable name to some directly routed names.

It is also worth noting that mobility support solutions are also potential routing scalability solutions [14]. In IP, examples include Mobile IP [15], which implements ocator/identifier separation, and Cellular IP [16], which utilizes per-hop trace to update the forwarding path from a reachable anchor to a mobile node, thus making the mobile node reachable. The Cellular IP work also hints a principally different direction for scaling routing: to handle dynamics by updating local forwarding states.

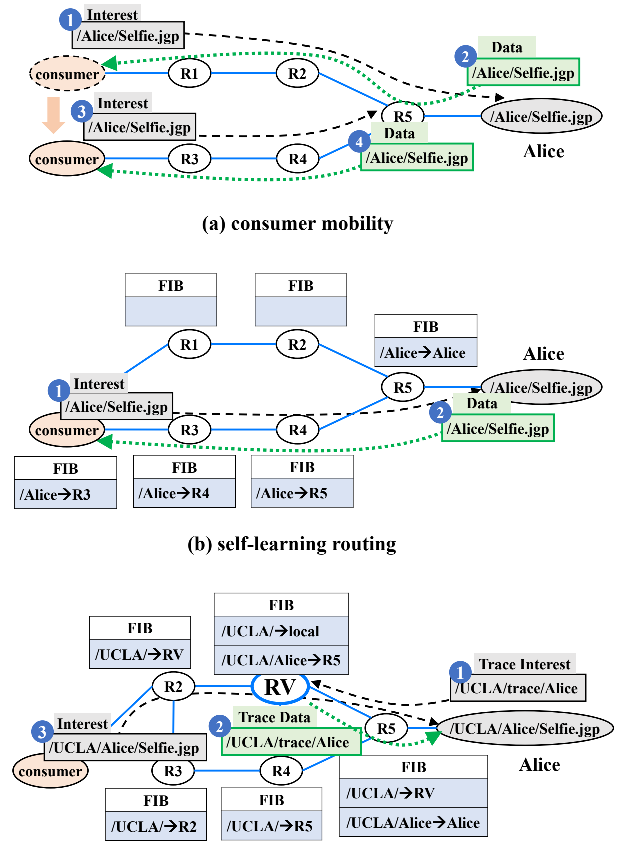
In summary, the basic approaches to routing scalability that have been developed for node-centric networks can also be applied to scale routing in data-centric networks. The routing scalability problem exists in both architectures, and the guiding principles for solutions are architecture-independent. The key



(a) map-n-encap scheme

(b) forwarding hint

Fig. 4. Mapping-based solutions in IP and NDN



(a) consumer mobility

(b) self-learning routing

(c) KITE

Fig. 5. Tracing-based solutions in NDN

design choice for naming/forwarding is which identifier (or its prefix) is announced into the routing plane, and how to provide the reachability to non-globally-routable identifiers.

IV. What Changes

In this section, we identify the factors related to routing scalability that are changed by the architectural transition from node-centric to data-centric model, and discuss several new routing scalability solutions that are enabled by naming data and NDN's stateful forwarding plane. As discussed in [17], because of the stateful forwarding plane, freeing routing protocols from short-term churns can greatly improve their scalability, enabling NDN to use routing protocols that were previously viewed as unsuitable for real networks, such as hyperbolic routing [18]. Below we introduce some routing-independent solutions for scalability.

A. Mapping-Based Solutions

One category of routing scalability solutions is to use a mapping system to map identifiers that are not injected into routing announcements to those that are announced to globally routing system, making the routing plane responsible only for a portion of all the identifier space which is stable and aggregatable, leading to smaller FIB size and fewer routing updates.

An early proposal of mapping-based solution for IP networks, map-n-encap [9] is illustrated in Fig. 4 (a). When Packet-1 reaches a Default Free Zone (DFZ) router EA, the destination address is mapped to a DFZ egress router address IP_EB. Then the original packet is encapsulated to create Packet-2 with a destination address of the egress router. Upon receiving Packet-2, the egress router EB removes the encapsulating header and forward the original Packet-1 to the destination.

SNAMP [19] is a mapping-based solution proposed for NDN, as shown in Fig. 4 (b). A consumer can obtain the whereabout of the desired data, i.e. the data locator, from a mapping service, but instead of encapsulating the Interest packet (which results in changing the name carried in the Interest), SNAMP lets an Interest packet carry the locator as a *forwarding hint*. Different from an IP packet which can only carry a single identifier, an Interest packet (Packet-3) can carry both "what" (data name) and "where" (forwarding hint) to retrieve the data. While routers forward the Interest towards where data is available based on the forwarding hint, the data name remains visible to intermediate routers, so that they may match the Interest to cached data, without the Interest necessarily reaching the "where" location. Moreover, an Interest can be resent carrying different forwarding hints, an option at the discretion of data consumers.

B. Tracing-Based Solutions

Tracing-based solutions to routing scalability can thrive on NDN's stateful forwarding plane, which can be leveraged to create a "breadcrumb trail" to a data producer. In NDN forwarding, when a router receives an Interest and does not have the desired data, the router records the Interest with its incoming interface in the Pending Interest Table (PIT) and further forwards it based on the FIB. The states recorded in the PIT of each router traversed by the Interest create a reverse path ("breadcrumb trail") for the requested data to return. Although similar approaches were first introduced for

IP multicast [20] and IP mobility [16] support, the concept of creating state to form "breadcrumb trail" is at odds with IP's stateless data plane, thus they are not widely deployed.

One key feature of stateful forwarding plane is keeping consumers of FIBs. Because Data packets are returned by traversing the Interest (Packet 1) back to the consumer, consumer mobility is natively supported. As shown in Fig. 5 (a), when the consumer moves before Data packet (Packet 2) returns, the consumer re-expresses unsatisfied Interest (Packet 3) which creates/updates the reverse path back to its current location.

Self-learning routing [21] uses the stateful forwarding plane to establish consumer-initiated trace to the producer. As shown in Fig. 5 (b), in the absence of a matching FIB entry, an Interest packet (Packet 1) can be flooded or randomly unicast to eventually reach the producer. On the Data packet's (Packet 2) way back to the consumer, each router along the trace creates a FIB entry for the corresponding data prefix, pointing to the incoming interface of the Data packet.

KITE [22] is a tracing-based producer mobility solution. KITE enables data retrieval from a mobile producer by setting up a path in FIBs from a stationary rendezvous server (RV) to a mobile producer. As shown in Fig. 5 (c), the RV makes routing announcement for a shorter data name prefix ("/UCLA"). The producer Alice issues a signed Trace Interest (TI) (Packet 1) to the RV. The RV verifies the TI and responds with a signed Trace Data (TD) (Packet 2), which then travels back to the mobile producer. Upon receiving a TD, intermediate routers (RV and R5) create or update FIB entries for the data name prefix ("/UCLA/Alice") pointing to the incoming interfaces of the TI. A consumer Interest (Packet 3) will be forwarded toward the RV to reach the producer. We note that the data name prefix ("/UCLA/Alice") only appears in FIBs along a trace, removing the burden of the routing plane from tracking mobile prefixes.

C. Data Rendezvous

By focusing on fetching named data, data-centric networking opens an entirely new dimension to routing scalability: *data rendezvous*, which captures the concept that Interests simply need to meet data at some rendezvous point, which does not have to be the original data producer. In-network caching (Fig. 6 (e)) exemplified this concept: popular data automatically get pulled away from its producer and distributed to multiple router caches which are closer to potential consumers. In node-centric networks, network layer can only identify nodes, application data caching can only be done at application layer.

Data rendezvous in NDN can be achieved by moving data produced by mobiles to stationary servers that are already routable. We call those servers "data depot" [23] (Fig. 6 (f)). A data depot may look similar to home agent in Mobile IP or the RV server in KITE, except that data depot takes the full responsibility for hosting the data instead of simply assisting Interest forwarding. Once user data is uploaded to the depot, the data can be easily retrieved by others. A data depot may

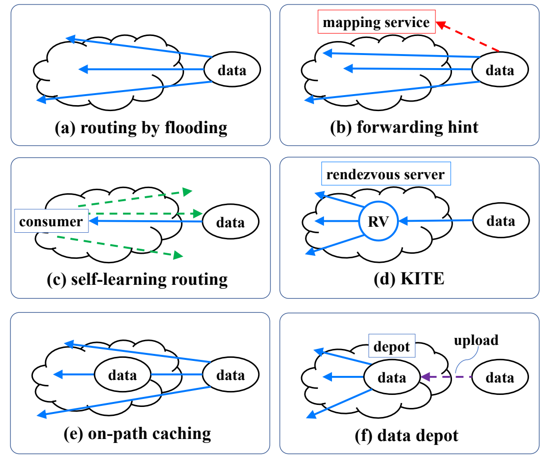


Fig. 6. Different solutions to achieve the reachability

seem equivalent to a known mapping server, except that it already has the data one wants. Today's cloud storage services, e.g., Dropbox, Google Drive, conceptually play the role of a data depot, but require users to connect to specific nodes first.

The native support for consumer mobility and tracing-based support for producer mobility make NDN mobility-friendly. If all the data produced by personal devices and Internet-of-Things sensors are collected to data depots, then device-specific prefixes do not need to enter the routing plane. However, since there is a cost of getting data into data depots, a new research question is to understand different data rendezvous design choices, including a) collecting data availability but not moving data, b) moving data to edge storage [24], and c) moving data to a data depot.

In summary, network forwarding requires reachability information to all identifiers. Fig. 6 illustrates how the solutions introduced above achieve this goal, demonstrating the changes introduced by data-centric networking.

V. Related Work

In the early days, Shoch [25] defined the following terminologies: a "name" identifies "what", an "address" identifies "where", and a "route" identifies "how to get there". Saltzer [26] identified four types of objects that may be network destinations: service and users, nodes, network attachment points, and paths. Saltzer's view unified Shoch's definitions: a "name" in Shoch's definition is the name of a service that is usually meaningful to human, and the "address" of the service is the name of the node on which the service runs, and a "route" to the "address" is simply a path to a network attachment point of the node. Day [27] further developed the use of abstraction and binding to model node-centric networking. He believed that, in principle, communication happens between entities at the same layer. Day emphasized that the root cause of IP routing scalability is due to missing a layer that identifies hosts/nodes. In this paper, we clarify that when a name is directly used in routing/forwarding, then the name is the locator for what it refers to.

In investigating the design space of mobility support, Zave and Rexford [28] proposed a geomorphic model consisting of indefinite number of layers at various levels. DRM (Dynamic-routing Mobility) preserves an entity's identity when its location (i.e., attachment at network layer) changes, by updating intra-layer states, i.e., routes in the form of mappings from destination to next hop; SLM (Session-Location Mobility) changes the identity of a relocated entity to match the location change, and preserves an entity's identity at a higher layer by changing inter-layer mappings. We believe that tracing built on NDN's stateful data plane is a new scheme in the DRM category, forwarding hint enables a packet to carry both data name and locator in the SLM category, and data rendezvous represents a new solution beyond SLM and DRM. We refer readers to [23] for more details of mobility support in NDN.

Gao et al. [3] distinguished three "puristic" approaches for providing location independence: "indirection routing", "name-to-address resolution", and "name-based routing". The paper compared the three approaches quantitatively in terms of path stretch and update cost in device and content mobility scenario. Chaganti et al. [29] followed the steps of [3], and evaluated the three approaches in various parameterized mobility models. It is worth noting that in our categorization, both "indirection routing" and "name-to-address resolution" schemes belong to mapping-based approaches in the sense that both require a separate locator that represents the device/content's current location.

The term "location-independent architecture" deserves further clarification. While proposed network architectures like NDN [1] and MobilityFirst [30] each features a distinct networking paradigm, NDN uses application names at network layer, MobilityFirst clearly separates application namespaces from network address namespace. Since NDN names identify data chunks directly, NDN inspires a new direction of realizing location-independent communication that centers around data availability instead of node/channel availability.

VI. Concluding Remarks

We believe that the future of networking lies in recognizing the right communication abstraction. Data-centric networking makes a fundamentally different abstraction of communication model than today's node-centric IP networking. Opposite to a common concern regarding NDN routing scalability, NDN leads to brand new and more effective ways to realize existing solutions, namely namespace separation and mapping; it also further expands the solution space by enabling tracing and data rendezvous.

References

[1] L. Zhang, A. Afanasyev, J. Burke, V. Jacobson, P. Crowley, C. Papadopoulos, L. Wang, B. Zhang *et al.*, "Named data networking," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, 2014.

[2] Y. Zhang, L. Wang, A. Afanasyev, and L. Zhang, "Similar yet different: Protocol design choices in is-is and ospf," in *Proceedings of the Asian Internet Engineering Conference*. ACM, 2017, pp. 24–30.

[3] Z. Gao, A. Venkataramani, J. F. Kurose, and S. Heimlicher, "Towards a quantitative comparison of location-independent network architectures," in *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 4. ACM, 2014, pp. 259–270.

[4] A. L. Dul, "Global ip network mobility using border gateway protocol (bgp)," *Boeing White paper*, vol. 10, 2006.

[5] D. Meyer, L. Zhang, and K. Fall, "Report from the iab workshop on routing and addressing," Tech. Rep., 2007.

[6] H. Balakrishnan, K. Lakshminarayanan, S. Ratnasamy, S. Shenker, I. Stoica, and M. Walfish, "A layered naming architecture for the internet," in *ACM SIGCOMM Computer Communication Review*, vol. 34, no. 4. ACM, 2004, pp. 343–352.

[7] R. Moskowitz, P. Nikander, P. Jokela, and T. Henderson, "Host identity protocol," Tech. Rep., 2008.

[8] R. Atkinson and S. Bhatti, "Identifier-locator network protocol (ilnp) architectural description," Tech. Rep., 2012.

[9] S. Deering, "The map & encap scheme for scalable ipv4 routing with portable site prefixes," *Presentation, Xerox PARC*, 1996.

[10] M. O'Dell, "Gse: An alternate addressing architecture for ipv6. draftietf-ipngwg-gseaddr-00. txt," *Network Working Group*, 1997.

[11] D. Massey, "A proposal for scalable internet routing & addressing," *INTERNET DRAFT, draft-wang-ietf-00. txt*, 2007.

[12] D. Farinacci, V. Fuller, D. Meyer, and D. Lewis, "The locator/id separation protocol (lisp)," Tech. Rep., 2013.

[13] D. Massey, L. Wang, B. Zhang, and L. Zhang, "A scalable routing system design for future internet," in *Proc. of ACM SIGCOMM Workshop on IPv6*, 2007.

[14] E. Lear and R. Droms, "What's in a name: Thoughts from the nsrg," *Work in Progress*, 2003.

[15] C. E. Perkins, "Mobile ip," *IEEE communications Magazine*, vol. 35, no. 5, pp. 84–99, 1997.

[16] A. G. Valkó, "Cellular ip: A new approach to internet host mobility," *ACM SIGCOMM computer communication review*, vol. 29, no. 1, pp. 50–65, 1999.

[17] C. Yi, J. Abraham, A. Afanasyev, L. Wang, B. Zhang, and L. Zhang, "On the role of routing in named data networking," in *Proceedings of the 1st ACM Conference on Information-Centric Networking*. ACM, 2014, pp. 27–36.

[18] V. Lehman, A. Gawande, B. Zhang, L. Zhang, R. Aldecoa, D. Krioukov, and L. Wang, "An experimental investigation of hyperbolic routing with a smart forwarding plane in ndn," in *2016 IEEE/ACM 24th International Symposium on Quality of Service (IWQoS)*. IEEE, 2016, pp. 1–10.

[19] A. Afanasyev, C. Yi, L. Wang, B. Zhang, and L. Zhang, "Snamp: Secure namespace mapping to scale ndn forwarding," in *Computer Communications Workshops (INFOCOM WKSHPS), 2015 IEEE Conference on*. IEEE, 2015, pp. 281–286.

[20] D. Estrin, D. Farinacci, A. Helmy, D. Thaler, S. Deering, M. Handley, V. Jacobson, C.-G. Liu, P. Sharma, and L. Wei, "Protocol independent multicast-sparse mode (pim-sm): Protocol specification," Tech. Rep., 1998.

[21] J. Shi, E. Newberry, and B. Zhang, "On broadcast-based self-learning in named data networking," in *IFIP Networking Conference (IFIP Networking) and Workshops, 2017*. IEEE, 2017, pp. 1–9.

[22] Y. Zhang, Z. Xia, S. Mastorakis, and L. Zhang, "Kite: Producer mobility support in named data networking," in *5th ACM conference on Information Centric Networking (ICN 2018)*, 2018.

[23] Y. Zhang, A. Afanasyev, J. Burke, and L. Zhang, "A survey of mobility support in named data networking," in *Computer Communications Workshops (INFOCOM WKSHPS), 2016 IEEE Conference on*. IEEE, 2016, pp. 83–88.

[24] I. Psaras, O. Ascigil, S. Rene, G. Pavlou, A. Afanasyev, and L. Zhang, "Mobile data repositories at the edge," in {USENIX} *Workshop on Hot Topics in Edge Computing (HotEdge 18)*, 2018.

[25] J. F. Shoch, "Inter-networking naming, addressing, and routing," in *17th IEEE Comput. Soc. Int. Conf.(COMPCON)*, 1978, pp. 72–79.

[26] J. Saltzer, "On the naming and binding of network destinations," Tech. Rep., 1993.

[27] J. Day, *Patterns in network architecture: a return to fundamentals*. Pearson Education, 2007.

[28] P. Zave and J. Rexford, "The design space of network mobility," *Recent Advances in Networking. ACM SIGCOMM*, 2013.

[29] V. Chaganti, J. Kurose, and A. Venkataramani, "A cross-architectural quantitative evaluation of mobility approaches," in *IEEE INFOCOM*. IEEE, 2018, pp. 639–647.

[30] A. Venkataramani, J. F. Kurose, D. Raychaudhuri, K. Nagaraja, M. Mao, and S. Banerjee, "Mobilityfirst: a mobility-centric and trustworthy internet architecture," *ACM SIGCOMM Computer Communication Review*, vol. 44, no. 3, pp. 74–80, 2014.