ARL-TN-0845 ● SEP 2017

US Army Research Laboratory

Coexistence of Named Data Networking (NDN) and Software-Defined Networking (SDN)

by Vinod Mishra

ARL-TN-0845 ● SEP 2017

US Army Research Laboratory

Coexistence of Named Data Networking (NDN) and Software-Defined Networking (SDN)

by Vinod Mishra
Computational and Information Sciences Directorate, ARL

REPORT DOCUMENTATION PAGE			*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED (From - To)**
September 2017	Technical Note	January 2017–June 2017

4. TITLE AND SUBTITLE	**5a. CONTRACT NUMBER**
Coexistence of Named Data Networking (NDN) and Software-Defined Networking (SDN)	
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
6. AUTHOR(S)	**5d. PROJECT NUMBER**
Vinod Mishra	
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	**8. PERFORMING ORGANIZATION REPORT NUMBER**
US Army Research Laboratory ATTN: RDRL-CIH-S Aberdeen Proving Ground, MD 21005	ARL-TN-0845
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	**10. SPONSOR/MONITOR'S ACRONYM(S)**
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Software-Defined Networking (SDN) is a rapidly emerging technology that brings new flexibility to networking through the separation of the Control and Data Planes. At the same time, Named Data Networking (NDN) is another approach focusing on the information content of the object and not on its packet nature. This report explores the viability of coexistence and integration of NDN and SDN in the same tactical network and discusses its advantages and disadvantages.

15. SUBJECT TERMS

tactical networks; mobility; Software-Defined Networking; Named Data Networking, SDN controller

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Vinod Mishra
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	UU	20	**19b. TELEPHONE NUMBER (Include area code)**
Unclassified	Unclassified	Unclassified			(410) 278-0114

Standard Form 298 (Rev. 8/98)
Prescribed by ANSI Std. Z39.18

Contents

List of Figures

List of Tables

1. Introduction

It has become very clear now that the behavior of the end-users regarding the Internet has changed. They are mostly interested in using it for retrieving information (articles, pictures, videos, etc.) and have no preference as to the origin of that information. The current Internet was built under a different assumption, which was more like that of a telephone network. It assumed that a packet would be routed according to its IP address without any regard for the nature and subject matter of its payload. This architecture is not suitable for the content-centric behavior of the users.

Recently, the idea of Information- or Content-Centric Networking (ICN or CCN) has been advanced with the goal of overcoming its shortcomings related to information delivery. Named Data Networking (NDN) is one of the many realizations of this basic idea. Software-Defined Networking (SDN) is another innovative approach toward a programmable network based on the separation of the Control and Data Planes. In this report, we explore the possibility of their coexistence so the advantages of both can be realized in a single unified approach.

2. An Overview of NDN and Its Comparison with SDN

2.1 Basic SDN Principles

SDN is the new networking paradigm that is remaking traditional networking and communication in its image. It arose to bring the same level of flexibility in networking as has previously been available for computing. It brings 2 very important functionalities:

1) Data and Control Plane Separation: At a high level, the network performs 2 kinds of operations on the incoming packets. The control functions make changes in the packet header based on some policy, and data functions involve forwarding the packet to its destination. In traditional networks, both of these capabilities are exercised by the network device running vendor-proprietary software. This arrangement is very static and inflexible. In the SDN, all the control functions are exercised by a centralized controller and network devices only perform the forwarding functions. This separation of Data and Control Plane is the key to the new SDN paradigm.

2) Programmability: The previous separation makes it possible to program the Control Plane to make it responsive to automatic control and dynamic situations. The policies and applications in principle can be translated to

Control Plane commands and the network can respond to new situations in real time.

2.2 SDN Planes and Operations

At a high level, the networks supporting SDN have 3 planes:

1) Applications and Management Plane: The policy, management, and applications interact with the SDN controller through a North Bound Interface (NBI). Their intents are translated to desired control commands by the NBI. At this time, this interface has not been standardized.

2) Control Plane: The SDN Controller (SDNC) at this plane is a centralized entity that translates the application layer commands into appropriate switch operations and topologies. Currently there are many such SDNCs in existence based on different programming languages and internal organizations of the components. Some of the more frequently used include Open Day Light (ODL), Ryu, POX, and NOX. The South Bound Interface (SBI) of the SDNC has been standardized and is called OpenFlow (OF). It is used to control the network devices. The SDNC also collects the network state monitoring data and status information.

3) Data Plane: This plane consists of network devices, which can be both hardware and software. The most well-known among them are the OF compliant switches. It is also possible to have a hybrid Data Plane consisting of both OF and non-OF switches. The OF switches contain a Flow Table (FT), which has protocol-dependent entries specifying the source and destination end-points and other relevant information in its rows. For instance, the entries of source, destination, and match-action are common to many protocols. The main function of the OF switch is to forward the incoming packet to the next hop according to the FT entries.

The basic SDN operations start after the first end-user packet arrives at the OF switch. The switch matches the packet flow information to the FT entries. There are 2 possible scenarios.

1) Scenario 1: The incoming packet information is matched to an existing FT row.

In this case the switch forwards the packet through the port indicated by the matching entry row.

2) Scenario 2: The FT finds no matching entry.

The switch sends that packet to the SDNC using the OF protocol. The SDNC determines the new rules for forwarding the packet and sends this information to the switch. The switch adds a new row to the FT and forwards the packet to the next hop.

2.3 NDN Principles

The NDN is one of the Future Internet Architectures initiated by NSF. The main idea is to make named data the main focus of the networking communication process. It is based on the following basic principles:

1) The NDN repurposes the network layer to focus on the named data objects instead of the IP addresses. It thus does not constrain the growth and development of the higher and lower layers of the design stack.
2) It retains and expands the end-to-end design principle of the Internet.
3) It also retains the separation of routing and forwarding functions of the original Internet design. For that, it uses the best available forwarding schemes while it also develops new routing protocols.
4) It provides flow-balanced data delivery for stable network operation.
5) It enables end-user empowerment and open competition.
6) It provides a built-in security of the named data by requiring that all data must be signed by the producer. It is explained in detail below.

2.4 Built-in Security in NDN

In NDN, each data piece is secured by default using signature from data producer. The signature is mandatory and it binds the name with data. Together with the publisher's information, it creates a trust model in which the trust in data is decoupled from its origin and manner of creation. In NDN, the cryptographic key itself can be treated as a named data so the key distribution problem has a good solution. Additionally, requiring signatures on network routing and control messages can provide routing protocol security as well.

2.5 Names in NDN

In NDN, data producers and consumers can agree on a convention to name data in a structured and hierarchical manner. This can be unique to an application and a deterministic algorithm can be used so that both producers and consumers arrive at the same name for a desired piece of data. As the routers do not know names, the application-specific naming conventions can grow and evolve independently. Consumers also have the ability to retrieve data based on partial names based on the naming hierarchy. This approach can support capabilities in NDN like content distribution, multicast, mobility, and delay tolerant networking.

2.6 NDN Operation

The simple change of using data names at IP layer leads to profound changes in the data delivery operations in the network.

1) The end-user or data-consumer sends out an ***Interest Packet*** containing the desired data's name to the nearest NDN router.

2) Scenario 1: The interface through which the Interest Packet arrives at the router was also used earlier for the same data request

   The name-based routing protocol used by the router has already created a record in its ***Forwarding Information Base (FIB)***, which knows the next node to which the earlier packet was sent. It forwards the Interest Packet to that node. This process is repeated at the next node and finally the Interest Packet reaches the destination node containing the requested data. That ***Data Packet*** has both the name and the content being sought and is sent back by the same route it was reached.

3) Scenario 2: There is no previous record and the Interest Packet is new.

   All unfulfilled Interest Packets are stored in the router in a ***Pending Interest Table (PIT)***. Each PIT entry contains the name of the Interest and a set of interfaces from which the Interests for the same name have been received. In case of multiple Interests for the same data arriving in the router, only the first one is sent toward the data source. On receiving the Data Packet, the router finds the matching PIT entry and forwards the data to all the listed interfaces. After that, it removes that PIT entry, and caches the Data in the ***Content Store (CS)*** to satisfy any similar future requests. Because one Data satisfies one Interest across each hop, an NDN network achieves hop-by-hop flow balance. In case of multiple Interests for the same Data, only one flow moves across the network and this saves energy and resources.

Table 1 provides a comparison of SDN and NDN.

Table 1 SDN-NDN comparison

Functions	Support in SDN	Support in NDN
Information Retrieval	No support for named data retrieval	Yes, this support is NDN's defining attribute
Multidomain Extension or Scalability	Multidomain controller architecture with East-West Bound Interfaces. A single domain scaling is dependent upon the controller processing power.	Yes, using BGP connection between domains like current networks
Names	No support	Well-developed approach to name semantics
Security	Many non-SDN security problems absent. Some new attack vectors become possible.	Name-based security, more robust than endpoint-based security, some new security threats emerge such as Interest flooding
Routing and Forwarding	Flow based and centrally managed.	Name based. Each node has control logic for routing
Protocol support	Version dependent, POF and P4 may be solutions	New NDN protocols in addition to older ones
Data Plane Intelligence	No control intelligence	Intelligent
Caching	Only allowed through FT entries	Automatic caching enabled, Buffer data reusable
Privacy	As strong as current networks	Requester privacy guaranteed
Transport		

3. State of the Art in the Integration of CCN/ICN with SDN

The enterprise networks have stable and static topologies and ICN ideas have been tested there more extensively. There have been very few initiatives exploring the coexistence of SDN and NDN[1]. On the other hand, there have been some studies integrating SDN with other versions of ICN. Some recent examples are described here:

1) coCONET

European Union started the now concluded OpenFlow in Europe: Linking Infrastructure and Applications (OFELIA) program.[2,3] earlier. Its offshoot, the Content Network (CONET) project[4–7] was the first one to combine the principles of the basic ICN with SDN as shown in Fig.1.

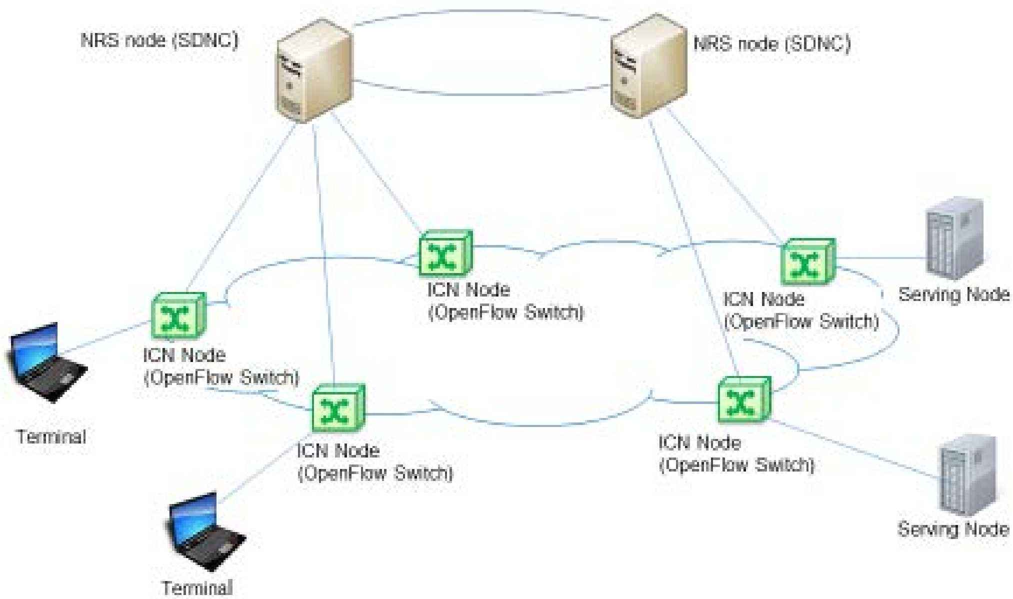

Fig. 1 coCONET topology

In CONET, the named data are represented as composed of one or many logical components called chunks. An Interest Packet sent by the user contains information about the name and the chunk pair. The CONET node lacks the intelligence to resolve this information into routing instruction, so it queries the Name Routing System (NRS) for finding the next hop route for a given name-chunk pair. This is different from NDN in which each node has the ability to find routing using the Forwarding Strategy Module (FSM) and FIB.

The coCONET work initially used OpenFlow 1.0 switches and used the options field of the IP packet field to demonstrate the basic idea of named data retrieval. Reference 3 also puts forward an architecture in which the NRS module is seen to be a part of the SDN Controller. In this vision, the OpenFlow protocol has to be extended to accommodate the name data addressing system.[8,9] One can envision a future in which a standardized ICN named-data scheme can be included in a future version of the OpenFlow protocol. In its absence, a Protocol Oblivious Forwarding (POF)[10] approach may be useful.

2) ContentFlow

This is another significant effort[11] from WinLab integrating SDN with ICN. It tries to use minimum extension to the existing SDN network by adding content-dependent flow-headers. Here the SDNC 1) manages the content, 2) resolves content to location, 3) enables content-based routing and forwarding policies, 4) manages the content caching, and 5) allows SDNC to create new content-based network mechanisms. There is a new network element of "Cache", which interfaces with the controller. All these are shown in Fig. 2.

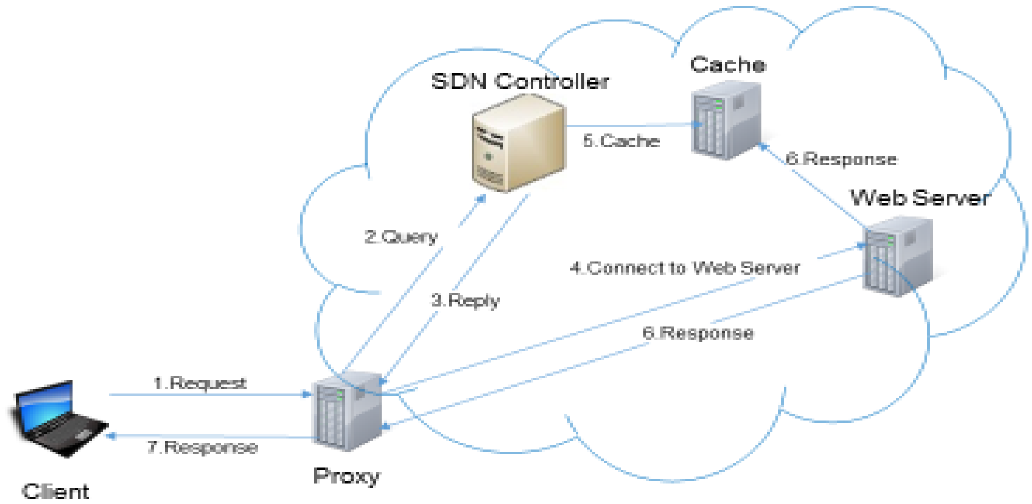

Fig. 2 ContentFlow topology

Functions supported by each technology, redundancies, strengths, weaknesses, and gaps:

We come back to the problem of comparing NDN and SDN networks with respect to various networking functionalities.

3) CCNx with Open vSwitch and Floodlight

Li et al. [12] focus on modifying the OpenFlow protocol, and use SDN functions to enable ICN. Their design is based on CCNx, Floodlight controller, and a revised Open vSwitch (a virtual switch in SDN). This testbed is claimed to do the following:

a) Allow the identification of the content requests,

b) Realize the routing process of ICN in SDN,

c) Deliver content efficiently, and

d) Enable deployment in existing IP networks.

In addition, the proposed approach continues to support traditional host-to-host communication using TCP/IP.

4) SDICN

Wang et al.[13] combine SDN architecture with essential ICN ideas. They have developed 3 new algorithms for this purpose:

a) Content Locating (CL) for redirecting user request to needed data,

b) Content Optimal Deployment (COD) for deploying the data found by CL at optimal node, and

c) Path Optimizer (PO) for balancing network traffic in both spatial and temporal dimensions.

According to the paper, their system outperforms CCNx.

5) SD-ICN

Wang et al.[14] from City University of Hong Kong and Huawei deploy Open vSwitch supporting OpenFlow to add ICN over the SDN network. Their main contributions are as follows:

a) Abstraction of common function modules (e.g., content distribution management and name-to-tag mapping) for deploying and inter-operating different ICN architectures.

b) Definition of a unified packet tagging scheme to forward packets of different ICN architectures over the same physical network.

c) Extension of the following:

- The OpenFlow interface to disseminate cache and interoperability related decisions.
- The Open vSwitch to tag the original ICN packets at the arrival point of the ingress routers and to untag them at the arrival point of the egress routers.
- The flow tables to support cache-related operations including shared cache management.

6) SDN and ICN with Big Data

Yao et al.[15] combine CCN, SDN, and Big Data processing in a unified framework and call it Data-Driven Networking (DDN). They also simulate a particular use case in the DDN framework.

4. A Proposed SDN-NDN Integration Scheme

The NDN nodes contain both the Control and Data Plane functions and fit very well with the distributed networking approach. It has to be re-architected to support the SDN paradigm.

4.1 SDN-NDN Integration for Networks with Fixed Infrastructure

The general enterprise networks have fixed nodes like routers, switches, and optical nodes. The integrated NDN-SDN architecture will have new distribution of functions among the traditional SDN planes. These distributions are as follows:

1) *Application and Management Plane*: It remains the same as before except the SDNC NBI is able to handle NDN-specific aspects of the application and management.

2) *Control Plane*: SDNC now contains FSM and the CS. This removes the main control functions from standard NDN nodes and puts them in SDNC. The CS stores the contents retrieved due to user Interests and its placement in the SDNC removes the large storage need of the regular NDN nodes.

3) *Data Plane*: The new integrated node contains the FT, PIT, and FIB modules. This scheme expands the function of the OF switches. The new nodes will have larger buffers for caching the table data.

The new routing protocol Named data Link State Routing protocol (NLSR) functionality can be integrated with SDN with moderate effort.

4.2 SDN-NDN Integration for Networks with Ad Hoc Infrastructure

The mobile ad-hoc networks (MANETs) are formed from mobile devices in response to some mission requirements. They have unstable and dynamic topologies and can be characterized as being disrupted, intermittent, and latency-prone (DIL). Their physical layer is usually in flux and sudden topology changes are very common. Currently, MANET networks are IP-based. It is desirable to bring advantages of both SDN and NDN to this environment. There are many challenges originating from the specific capabilities of MANET, NDN, and SDN technologies, which make it a challenge to integrate them in a coherent manner.

The main challenge is again the separation and placement of the network functions among the Control and Data Planes. The Application and Management Plane remains relatively unchanged. The planes at this point are as follows:

1) *Application and Management Plane*: It retains the previous integration of SDNC NBI and NDN as in the fixed infrastructure case. MANET itself does not cause changes to this layer except through changes in the applications and management functions due to the new physical environment.

2) *Control Plane*: FIB and CS still remain at this plane. In addition, the SDNC has to retain the last state of the network device so that in case of broken connection it can remember the node properties to reengage it in the process. It will also have the capability to distribute policies to the nodes after they are received from the top layer.

3) *Data Plane*: The new integrated node contains the FT, PIT, and FIB modules as before. It will also be capable of neighbor discovery in case of broken connection.

5. Conclusions

SDN and NDN can be complementary, but much research still needs to be carried out to find their optimal integration architecture. The benefit of this integration will be helpful to the Warfighter's communication.

6. References

1. van Adrichem NLM, Kuipers FA. NDNFlow: Software-defined named data networking. IEEE International Conference on Network Softwarization (NetSoft 2015); 2015; University College London, UK.
2. Köpsel A, Woesner H. OFELIA – Pan-European test facility for OpenFlow experimentation. Lecture Notes in Computer Science. 2011;6994. Springer.
3. OFELIA project. [accessed 2017 June 28]. http://www.fp7-ofelia.eu.
4. Detti A, Blefari-Melazzi N, Salsano S, Pomposini M. CONET: A content-centric inter-networking architecture. Proceedings of ACM Sigcomm – ICN 2011; 2011 Aug; Toronto, Ontario, Canada.
5. Veltri L, Morabito G, Salsano S, Blefari-Melazzi N, Detti A. Supporting information-centric functionality in software defined networks. SDN'12: Workshop on Software Defined Networks, collocated with IEEE ICC; 2012 Jun 10–15; Ottawa, Canada.
6. Mazza G, Morabito G, Salsano S. Supporting content networking in OpenFlow. 2011 Nov. http://netgroup.uniroma2.it/CONET.
7. Blefari-Melazzi N, Detti A, Mazza G, Morabito G, Salsano S, Veltri L. An OpenFlow-based testbed for information centric networking. Future Network & Mobile Summit 2012; 2012 Jul 4–6; Berlin, Germany.
8. Chanda A, Westphal C. ContentFlow: Mapping content to flows in software defined networks, version 1. Ithaca (NY): Cornell University Library; 2013 Feb 6. arXiv:1302.1493.
9. Kotronis V, Dimitropoulos X, Ager B. Outsourcing the routing control logic: better internet routing based on SDN principles. 11th ACM Workshop on Hot Topics in Networks (HotNets-XI); 2012; New York, NY.
10. Li S, Hu D, Fang W, Ma S, Chen C, Huang H, Zhu Z. Protocol oblivious forwarding (POF): software-defined networking with enhanced programmability. IEEE Network Magazine. 2017;31:58–66.
11. Chanda A, Westphal C. ContentFlow: Mapping content to flows in software defined networks, version 2. Ithaca (NY): Cornell University Library; 2013 Feb 7. arXiv: 1302.1493v2 [cs.NI].

12. Li P, Muqing W, Ning W, Hongbao L. Supporting information-centric networking in SDN. Intl J Future Computer and Communication. 2015 Dec;4(6).

13. Wang X, Chen M, Hu C, Weng X, Xing C. SDICN: A software defined deployable framework of information centric networking. China Communications. 2016 Mar.

14. Wang J, Gao W, Liang Y, Qin R, Wang J, Liu S. SD-ICN: an interoperable deployment framework for software-defined information-centric networks. Poster at INFOCOM; 2014.

15. Yao H, Qiu C, Fang C, Chen X, Yu F. Richard. Content-centric and software-defined networking with big data. Ithaca (NY): Cornell University Library; 2016 Nov 19. arXiv: 1611.06299v1 [cs.NI].

List of Symbols, Abbreviations, and Acronyms

CL	Content Locating
COD	Content Optimal Deployment
CONET	Content Network
CCN	Content-Centric Networking
CS	Content Store
DDN	Data-Driven Networking
DIL	disrupted, intermittent, and latency-prone
FIB	Forwarding Information Base
FSM	Forwarding Strategy Module
FT	Flow Table
ICN	Information-Centric Networking
MANET	mobile ad-hoc networks
NBI	North Bound Interface
NDN	Named Data Networking
NRS	Name Routing System
ODL	Open Day Light
OF	OpenFlow
OFELIA	OpenFlow in Europe: Linking Infrastructure and Applications
PIT	Pending Interest Table
PO	Path Optimizer
SBI	South Bound Interface
SDN	Software-Defined Networking
SDNC	SDN Controller

1 (PDF)	DEFENSE TECHNICAL INFORMATION CTR DTIC OCA
2 (PDF)	DIR ARL RDRL CIO L IMAL HRA MAIL & RECORDS MGMT
1 (PDF)	GOVT PRINTG OFC A MALHOTRA
1 (PDF)	ARL RDRL CIH S V MISHRA