Hybrid SDN-ICN Architecture Design for the Internet of Things

Huda Saadeh[1], Wesam Almobaideen[2,3], Khair Eddin Sabri[3] and Maha Saadeh[4]
[1]*Dept. of Information Security University of Petra,* Amman, Jordan
Email: hsaadeh@uop.edu.jo
[2]*Dept. of Electrical Engineering and Computing Sciences, Rochester Institute of Technology*, Dubai, UAE
Email: wxacad@rit.edu,
[3]*Dept. of Computer Science, University of Jordan*, Amman, Jordan
Email: almobaideen@inf.ju.edu.jo, k.sabri@ju.edu.jo
[4]*Dept. of Computer Engineering and Informatics, Middlesex University*, Dubai, UAE
Email: m.saadeh@mdx.ac.ae

***Abstract*— Internet of Things (IoT) impacts the current network with many challenges due to the variation, heterogeneity of its devices and running technologies. For those reasons, monitoring and controlling network efficiently can rise the performance of the network and adapts network techniques according to environment measurements. This paper proposes a new privacy aware-IoT architecture that combines the benefits of both Information Centric Network (ICN) and Software Defined Network (SDN) paradigms. In this architecture controlling functionalities are distributed over multiple planes: operational plane which is considered as smart ICN data plane with Controllers that control local clusters, tactical plane which is an Edge environment to take controlling decisions based on small number of clusters, and strategic plane which is a cloud controlling environment to make long-term decision that affects the whole network. Deployment options of this architecture is discussed and SDN enhancement due to in-network caching is evaluated.**

Keywords*— *Cloud Computing, Edge/Fog Computing, IoT, ICN, SDN, Publish/Subscribe, OpenFlow, EWBI, NBI

I. Introduction

Internet of Things (IoT) is related to the world-wide connectivity of variant types of objects or devices, which communicate and cooperate to accomplish certain functionalities [1], [2]. The widespread of IoT causes new challenges over current IP networks such as mobility, scalability, interoperability, and privacy [3]. Technologies are evolving tremendously. For that reason, solutions should be smart, adaptive and dynamic to handle variations and challenges. One of the new networking paradigms that targets network modifiability, and maintenance is Software Defined Network (SDN) [4]. In SDN the network is separated as control and data planes. The data plane consists of data carriers, hosts, switch devices that direct the flow of data. The control plane is composed of controllers which are devices with upper level software that monitor and control the algorithms used in data routing, security, and provide network visualization such as OpenDaylight and Open vSwitch. This separation gives administrators and ISPs the capability to monitor, control network parties, adapt new network policies and network procedures. Consequently, results in controllable QoSs which are provided to users [4], [5], [6], [7]. Interaction between control plane and data plane is usually managed by southbound protocols such as OpenFlow (OF) [4].

Other Networking paradigms that can enhance the functionality of IP networking in the context of IoT are Content Delivery Network (CDN) and Information Centric Network (ICN). The former focuses on providing certain users application-specific caching services, such as Google caches. In CDN storage are used to cache users requests and services distributed in different locations to decrease traffic of communications with original servers. CDN has its own complexity and inherits IoT challenges over IP legacy-network. Moreover, CDN caches are application-specific and the service is provided only for contracted users [8]. ICN retrieves information, services, and actuation commands according to their names not according to their location [9], [10]. As well as, ICN gets the benefit of content holders or caches over the network to keep responses. Using in-network caches to store responses of user's requests enables subsequent requests to be served from cached copies closer to users rather than original sources, to decrease the delays of responses and the traffic load on the network [11]. ICN supports handling IoT challenges, mentioned before, and fits with the nature of many IoT applications because these applications are information retrieval by nature [12], [13].

Designers of network architectures should take into consideration the challenges that face current networks. One can observe the importance of combining the benefits of multiple networking paradigms such as ICN and SDN [14], [15] in facing these challenges. SDN controlling capabilities provides flexibility in deploying ICN architectures and enhances the management of several ICN main functions such as caching, routing, and naming resolution [15]. For instance, in [14] SDN controllers and switches has been used to predict the popularity of contents to enhance ICN caching. On the other hand, ICN is promising in the evolution of SDN functionalities to become content-based rather than flow-based [15]. Therefore, this research focuses on analyzing a set of literature proposed ICN architecture's designs that have separation of functionalities over multiple planes with different interaction interfaces [2], [16], [17] and similar to SDN layering approach. This paper also proposes a hybrid SDN-ICN architecture for IoT, expanding the ICN architecture proposed in [18]. The proposed architecture is conceptually evaluated, and different controlling scenarios are analyzed.

The rest of this paper is organized as follow: Section II reviews related work. A detailed explanation of the proposed hybrid SDN-ICN architecture is presented in Section III. Section IV provides a preliminary evaluation of deployment SDN with and without in-network caches. And Section V concludes the paper and presents future work.

II. Background and Related Work

This section presents background of SDN-ICN research and reviews related SDN-ICN architectures.

A. Background

The research field of designing and deploying ICN and SDN architectures has a lot of interest and efforts. Many proposals have been found in the literature that discuss and analyze the benefits that each of ICN and SDN can gain from each other. Some of these benefits are summarized as follow:

Benefits that ICN gains from SDN:

- SDN provides wide approaches for ICN deployment, which vary from clean-slate approaches to overlay approaches to cope with IP legacy networks and infrastructure [15], [19], [20].
- SDN control plane supports ICN in managing, enhancing, and implementing ICN core functions such as naming resolution, forwarding, caching, security, etc. [14], [20].
- Decoupling controlling, and data planes in SDN supports interoperability between variant ICN architectures, as well as, IP legacy infrastructure [19].

Benefits that SDN gains from ICN:

- In-network caching of ICN enhances the performance of the SDN network and decreases traffic [15], [20].
- Controlling network can be implemented content-based rather than traffic-based which enhances users' QoE [15], [21], [22].

Authors have used different methods in ICN deployment to be integrated with SDN and IP legacy devices and protocols according to many aspects, such as methods for naming and mapping, methods for routing, and methods for caching [15], [23]. In this paper some of these methods can be summarized as follow:

- Using the option fields in IPv4 and IPv6 protocols to store contents data, names, metadata, and tags [24].
- UDP packets to embed naming identifiers for ICN caching and forwarding identifiers to preserve compatibility with SDN OF protocol and other legacy devices [20].
- TCP Header reserved flags are used to distinguish between ICN and non-ICN packets [25].
- Imbedding information in HTTP headers and PDI [26].
- Building ICN sublayers over Transport layer to support ICN packets [25].
- Tagging and using tags-names-location mappings [27].
- Modifying and upgrading protocols such as OpenFlow-ICN based versions [20], HTTP request packets and DNS query packets [25].
- Using proxies, modules, Gateways, Special SDN-ICN based switches [20], [22], [26].

B. Related Work

The earlier ICN architectures concentrate on data delivery, caching, and content naming such as DONA [28], NDN [29]. These functionalities are implemented in the data plane of later architectures such as NetInf [30] and PSIRP [31] that added another controlling plane functionality. Controlling plane's functionalities provid more network monitoring capabilities and enhance management of ICN functions such as network - aware routing. The control plane in CONVERGENCE [17] (CoMid), controls contents creation, and consumption through publish/subscribe (pub/sub) approach. It also orchestrates set of CONVERGENCE data plane (CoNet) functionalities that perform caching, data delivery [32], and security functions [17]. Another pub/sub communication approach for control plane has been proposed in C-DAX [33] which is an ICN architecture that is designed for IoT. The control plane has been implemented in cloud with name resolution and security functions. Meanwhile data plane communications include mechanisms for data forwarding from data sources [33], [34].

A control plane, forwarding plane, and a data plane separation is found in COMET [16]. The data plane is built using CURLENG ICN architecture [35] with caching functions. The control plane in COMET (CMP) is responsible for name resolution, network monitoring to support the content-aware forwarding functionality of the Content Forwarding Plane (CFP) [16].

PPUSTMAN [18] is composed of data plane clusters that follow ICN paradigm. Caching, routing, naming resolution, forwarding, and security functions are managed by pub/sub controlling units supplied with central caches. There are three types of controlling units with different controlling functionalities: Controller, View, and Model. This pub/sub Model-View-Controller (MVC) design provides location and identity privacy for both publishers and subscribers. As well as, it enhances performance by adding levels of abstraction and concurrency between MVC nodes.

The architecture in [20] combines three blocks of network items: Edge network in which ICN servers collect information from sensor nodes and register them into the ICN cache servers through the forwarding block. The forwarding block consists of OpenFlow enabled ICN gateways which integrates between general ICN protocols and OpenFlow based ICN protocol to communicate with OF- ICN switches with caching capabilities. The OF-ICN controllers manages routing, caching, security, and naming through different managing modules.

Different SDN-ICN deployment options have been proposed and analyzed in [19]. The first option is a clean-slate pure ICN architecture. The second option uses static caches managed by a controller with special ICN software. The third option assumes edge nodes that forwards interests and data packets to the closest ICN capable nodes. The fourth option has a similar architecture as the third option with the difference of using tagging instead of tunneling. The fifth option uses tagging with dynamic ICN caches placement.

An overlay-ICN architecture [25], has separated the functionalities of naming, naming resolution, and routing into the control plane from the forwarding and caching functionalities which are handled in the data plane. The authors have presented two different deployment scenarios with SDN. The first one suggested the separation between ICN management and SDN controlling functionalities. While the second one suggested merging ICN management with SDN controlling functionalities. Similar SDN overlay approach is found in [22], where customer networks are communicating to ICN overlay network through ICN gateways. SDN controllers are used to manage name resolution and monitor requests that fails to find data. In [24] an overlay approach used the option field in the IP protocol to integrate ICN messages to SDN switches using names-location identifiers. And controllers are used to optimize data distribution in the network caches to load balance data traffic.

A summary of controlling plane functionalities in each architecture is presented in Table I.

TABLE I. SUMMARY OF CONTROL PLANE FUNCTIONALITIES IMPLEMENTED IN DIFFERENT ICN ARCHITECTURES

Arch.	**Functionalities**						
	C	***R***	***NRS***	***F***	***S***	***N***	***M***
[16]			√	√			√
[17]				√	√	√	
[18]	√	√	√	√	√		√
[20]	√	√			√	√	
[22]		√	√				√
[24]	√			√			
[25]		√	√			√	
[30]			√	√			√
[31]			√	√			√
[33]			√		√		

C: Caching, R: Routing Requests, NRS: Name Resolution, F: Forwarding Data, S: Security, N: Naming, M: Monitoring

Different SDN-ICN deployment options have been proposed and analyzed in [19]. The first option is a clean-slate pure ICN architecture. The second option uses static caches managed by a controller with special ICN software. The third option assumes edge nodes that forwards interests and data packets to the closest ICN capable nodes. The fourth option has a similar architecture as the third option with the difference of using tagging instead of tunneling. The fifth option uses tagging with dynamic ICN caches placement.

III. HYBRID SDN-ICN PROPOSED ARCHITECTURE

As observed in the previous section and up to our knowledge most of the SDN-ICN research focuses on deployment approaches and using SDN controlling capabilities to support ICN functions. This paper proposes a separation of controlling functions over multiple controlling planes: Operational, Tactical, and Strategic planes. Each controlling plane is composed of three controlling units: The Control unit, the View unit, and the Model unit. This proposal expands PPUSTMAN architecture proposed in [18].

A. Overall View of the proposed planes

The first plane in this new proposal is PPUSTMAN with its both data and control planes to be considered as smart data plane (Operational Plane). The pub/sub MVC controlling units enhance modifiability and allow deployment of adaptive policies in the network cluster they manage [36]. They also manage data delivery between publishers and subscribers who are decoupled in space and time. Users authentication, Contents registration, publishing, and subscription are managed by the Control unit which cooperates with the Model unit which manages the central caches in turn. Forwarding published data to subscribers are managed by the View units. The advantage of MVC communication pattern over three-Tier communication pattern is analyzed in [18]. Moreover, sending actuation commands scenarios are also verified using situation calculus. Later, location-privacy aware routing protocol is simulated using NS2 over PPUSTMAN. The Control units in Operational plane monitor their clusters for adapting the policies and techniques according to each cluster's state. They also provide reports periodically to upper layer's Control units.

The second plane is the Tactical plane. This plane is divided into set of clusters. Each cluster is responsible for the behavior of a set of Operational plane's clusters. Tactical plane consists of pub/sub MVC controlling units that collects information from Control units in the Operational plane. The Control units take short-term decisions for small numbers of network clusters. Tactical plane Controllers also perform data preprocessing such as data encryption, before sending data to the upper layer plane's Controllers.

The third plane is the Strategic plane. This plane is designed similarly to the Tactical plane, where each cluster is responsible for a set of underlaying tactical plane's clusters. This hierarchy provides the Strategic plane's controlling units with a whole picture of the network to perform long-term decisions. Strategic plane's MVC units implement tasks that require extensive computational capabilities, such as data analysis, visualization, AI extensive computations, intrusion detection, and others. On the other hand, tactical plane is responsible for smaller network areas, and performs less extensive processing.

Control units in the Operational, Tactical, and Strategic planes cooperate to monitor network's state, users' behavior, privacy requirements, threats, attacks, freshness and popularity of cached contents, and many other factors. According to these factors different applied algorithms and policies are periodically adapted and managed.

B. *SDN Communication Interfaces*

Figure 1 shows the SDN interfaces design to be used between different architecture parties. Tactical plane and Strategic plane Control units communicate using North Bound Interface (NBI). Tactical and Strategic plane's View units communicate with Operational plane's MVC units using North Bound Interface (NBI). Since MVC units in Operational plane control physical nodes, the interaction between View units and other physical nodes is designed to go through South Bound Interface (SBI) such as OpenFlow protocol. Inter-cluster interaction between MVC units in different clusters within each plane is designed to go through East West Bound Interface (EWBI).

C. *Cloud and Edge Computing for Hybrid SDN-ICN*

Cloud computing is used in systems as a centralized computing and storage environment with high shared capabilities. This environment is used to monitor distributed systems or devices, supports dynamic topologies, and supports mobility because of having overall view of architectures and nodes [37]. Cloud also can implement intensive processing tasks [38]. Some of the shortages of cloud architectures are responses delays and security risks which result from transferring and processing data far from their originator. Moreover, extensive data processing consumes resources of some IoT devices such as sensors [39], processing of data on cloud environment preserves resources, but it rises security risks. Therefore, Fog/Edge computing is widely used to support the performance of clouds by applying data preprocessing tasks closer to their originator, and if required, secures information before being sent to cloud [40], [41].

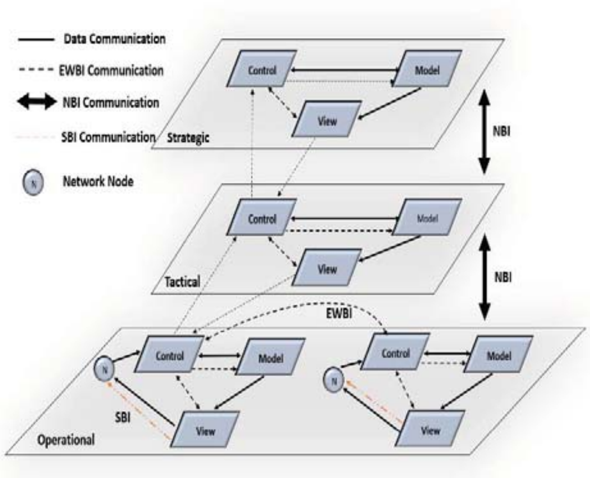

Fig. 1. SDN Interfaces between the three layers

In Hybrid SDN-ICN proposal, some SDN controlling functionalities is performed in Strategic plane which is deployed in cloud environment. To enhance performance and preserve security, Tactical plane is designed to be implemented using Edge computing. Controllers of Operational plane report information about their cluster's environment and unhandled issues to Tactical plane controllers. Tactical controllers handle some of these issues, while other issues that need more processing are preprocessed before being forwarded to the Strategic plane controllers.

Splitting controlling functionalities over multiple layers allows for the usage of less powerful controllers in Operational and Tactical planes than controllers in Strategic plane.

d. Deployment Options

Since inter-plane communications is occurred between controller units, SDN NBI protocols can be used without modifications. Similarly, SDN EWBI protocols can be used for inter-cluster communication and between MVC controlling units. In the Operational plane, there are four options to deploy the SDN-ICN functions:

- First, providing proxies to extract ICN information from HTTP and TCP headers to integrate with IP legacy nodes.
- Second, Adding ICN functionality for OF in OF-Switches to deal with content-based management.
- Third, View units which are responsible for delivering data and actions prepares data packets that is compatible with each subscriber's network. The mechanisms and software of View units can be updated as needed by upper level controlling units.
- Fourth, a clean-slate vision of SDN communication based on treating management commands from controllers similarly as ICN contents. Controllers publish actions into in-network caching, SDN switches and nodes subscribes to receive these actions periodically, and View units route these actions to their subscribers.

IV. SDN-ICN EVALUATION

In this paper, we evaluate the Fourth deployment option scenario to show the effect of SDN leveraging from ICN in-network caching. Other scenarios evaluation is left for future work.

The fourth scenario treats SDN-controlling commands similarly as IoT actuation commands. In this scenario SDN-switches register once as subscribers at control units for controlling actions retrieval. After authentication process, Control units update the registry tables at the Model units. Then each time the Control unit in the same plane or upper layer plane issues new actions for certain subscribers, The Model unit interacts with the View unit to deliver actions commands to some of the subscribers and keeping copies inside in-network caches to serve other neighbor subscribers. The following mathematical model shows how this scenario decreases actions packets traffic:

Assuming that each switch is located at distant M from the View unit, and S is the number of subscribed switches in the cluster and for simplification we assume that S=M. Equation 1 shows the worst-case traffic cost Tc to deliver actions to each switch from View without using caches and that M is the maximum distance in the cluster. Equation 2 calculates the average traffic cost to deliver action for all switches.

$$Tc = S*M = M*M \rightarrow O(M^2) \quad (1)$$

$$Tc = \frac{S\sum_{i=1}^{M} i}{M} = M(M+1)/2 \rightarrow O(M^2) \quad (2)$$

By providing caches capabilities to serve neighbor switches, View units do not have to deliver actions to each switch, on the other hand, they deliver copies to in-network caches. Since Caches placement and number affects the efficiency of the network, we assume that there is on cache and all switches are connected to it by one hop, for simplification. Equation 3 calculates the worst-case traffic cost in which the cache is located at maximum distance M. And Equation 4 calculates the average traffic cost taking into consideration that the cache is located at any distance from one hop to M hops far from View and all switches are assumed to be connected to this cache by one hop for simplicity.

$$Tc = S+M = M+M \rightarrow O(M) \quad (3)$$

$$Tc = \frac{\sum_{i=1}^{M} i}{M} = (M+1)/2 \rightarrow O(M) \quad (4)$$

Comparing the results of traffic cost in both scenarios shows to what extent SDN can be enhanced by ICN functions.

V. Conclusion and Future Work

Deployment ICN architectures based on SDN approach has been studied in literature. Many researches combined the benefits of both paradigms to enhance users QoE and to increase network performance [15]. This paper reviews some literature SDN-ICN architectures focusing on control plane functionalities in each. Based on that, we propose a Hybrid SDN-ICN architecture for IoT and discuss its deployment options.

The data plane of this architecture is an ICN architecture with control plane to perform and manage different ICN functions. Due to its direct effect on physical nodes this plane is considered as smart data plane or Operational plane. Additional higher-level controlling functionalities are proposed in Edge computing and Cloud computing environments. Tactical plane is deployed on edge nodes and is responsible for managing and monitoring a set of underlying clusters in the Operational plane. It also performs preprocessing tasks before sending data to the cloud for further processing. The Strategic plane is a cloud controlling environment which manages and monitor the whole network. It also performs long-term decisions for algorithms and policies adaptations.

Network clusters in each plane has three controlling units (Model, View and Control) rather than having all controlling functionality in one controlling unit (SDN controller). This separation of controlling functionalities over multiple units has number of benefits: First, it enhances the performance of controllers and allows for concurrent tasks processing as shown in [18]. Second, it provides more abstraction levels for nodes interactions. Third, it enhances interoperability between different paradigms and facilitates the deployment of ICN. The paper also evaluated the impact of having in-network caches on SDN actions disseminations, and it was found that traffic decreased significantly by ICN approach for delivering actions to switches.

In this paper we focused on discussing controlling functionalities deployment scenarios for this architecture leaving protocols upgrade and implantation for future work.

References

[1] S. Li, L. Da Xu and S. Zhao, "The internet of things: a survey," Information Systems Frontiers, vol. 17, no. 2, p. 243–259, 2015 b.

[2] H. Saadeh, W. Almobaideen and K. E. Sabri, "Internet of Things:A Review to Support IoT Architecture's Design," in 2nd International Conference on the Applications of Information Technology in Developing Renewable Energy Processes & Systems (IT-DREPS), Amman, Jordan, 2017.

[3] A. Lindgren, F. B. Abdesslem, O. Schelen, A. Malik and B. Ahlgren, "Applicability and Tradeoffs of Information-Centric Networking for Efficient IoT draft-lindgren-icnrg-efficientiot-00," 4 7 2014. [Online]. Available: https://tools.ietf.org/html/draft-lindgren-icnrg-efficientiot-00. [Accessed 25 4 2016].

[4] J. H. COX, J. Chung, J. CHUNG, S. DONOVAN, J. IVEY, R. J. CLARK, G. RILEY and H. L. OWEN, "Advancing Software-Defined Networks: A Survey," IEEE Access, vol. 5, pp. 25487 - 25526, 2017.

[5] S. Eum, M. Jibiki, M. Murata, H. Asaeda and N. Nishinaga, "A design of an ICN architecture within the framework of SDN," in Seventh International Conference on Ubiquitous and Future Networks, Sapporo, Japan, 2015.

[6] P. Li, W. Muqing, W. Ning and L. Hongbao, "Supporting Information-Centric Networking in SDN," International Journal of Future Computer and Communication, vol. 4, no. 6, 2015 a.

[7] Y. Zhang and Y. Wang, "SDN based ICN architecture for the future integration network," in 2016 16th International Symposium on Communications and Information Technologies (ISCIT), Qingdao, China, 2016.

[8] J. Suarez, J. Quevedo, I. Vidal, D. Corujo, J. Garcia-Reinoso and R. L. Aguiar, "A secure IoT management architecture based on Information-Centric Networking," Journal of Network and Computer Applications, vol. 63, pp. 190-204, 2016.

[9] AthanasiosV.Vasilakos, Z. Li, G. Simon and W. You, "Information centric network: Research challenges and opportunities," Journal of Network and Computer Applications, vol. 52, pp. 1-10, 2015.

[10] S. S. Adhatarao, J. Chen, M. Arumaithurai, X. Fu and K. K. Ramakrishnan, "Comparison of naming schema in ICN," in 2016 IEEE International Symposium on Local and Metropolitan Area Networks (LANMAN), Rome, Italy, 2016.

[11] M. F. Bari, S. R. Chowdhury, R. Ahmed, R. Boutaba and B. Mathieu, "A Survey of Naming and Routing in Information Centric-Networks," IEEE Communications Magazine, vol. 50, no. 12, pp. 44-53, 2012.

[12] B. Ahlgren, C. Dannewitz, C. Imbrenda, D. Kutscher and B. Ohlman, "A survey of information-centric networking," IEEE Communications Magazine, vol. 50, no. 7, pp. 26-36, 2012.

[13] M. Amadeo, C. Campolo, J. Quevedo, D. Corujo, A. Molinaro, A. Iera, R. L. Aguiar and A. V. Vasilakos, "Information-centric networking for the internet of things: challenges and opportunities," IEEE Network, vol. 30, no. 2, pp. 92 - 100, 2016.

[14] J. Z.-W. L. L.-X. P. J. C. Wai-Xi Liu, "Content Popularity Prediction and Caching for ICN: A Deep Learning Approach with SDN," IEEE Access, vol. 6, pp. 5075 - 5089, 2017.

[15] X.-W. W. M. H.-Q. L. S. K. D. QING-YI ZHANG, "Software Defined Networking Meets Information Centric Networking: A Survey," Accepted for publishing in IEEE Access, vol. 6, pp. 39547 - 39563, 2018.

[16] G. Pavlou, N. Wang, W. K. Chai and I. Psaras, "Internet-scale content mediation in information-centric networks," annals of telecommunications, vol. 68, no. 3-4, pp. 167-177, 2013.

[17] N. B. Melazzi, "CONVERGENCE: extending the media concept to include representations of real world," 20 1 2011. [Online]. Available: http://www.ict-convergence.eu/wp-content/uploads/C1.pdf. [Accessed 10 11 2017].

[18] H. K. Saadeh, W. Almobaideen and K. E. Sabri, "PPUSTMAN: Privacy-Aware PUblish/Subscribe IoT MVC Architecture Using Information Centric Networking," Modern Applied Science, vol. 12, no. 5, p. 128, 2018.

[19] M. Z. Robert Bauer, "Analysis of Deployment Options for SDN-based Information Centric Networking," in 2018 4th IEEE Conference on Network Softwarization and Workshops (NetSoft), Montreal, QC, Canada, 2018.

[20] M. J. M. H. A. N. N. Suyong Eum, "A design of an ICN architecture within the framework of SDN," in 2015 Seventh International Conference on Ubiquitous and Future Networks, sapporo,Japan, 2015.

[21] J. Z. ,. P. ,. S. Zhiguo Liu, "Satellite Network Architecture Design Based on SDN and ICN Technology," in 8th International Conference on Electronics Information and Emergency Communication (ICEIEC), Beijing, China, 2018.

[22] M. R. H. ,. I. C. ,. A. G. ,. G. B. Zeinab Zali, "A controller-based architecture for information centric network construction and topology management," China Communications, vol. 15, no. 7, pp. 131 - 145, 2018.

[23] L. C. F. Rihab Jmal, "Emerging Applications for Future Internet Approach Based-on SDN and ICN," in 2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA), Hammamet, Tunisia, 2017.

[24] C. M. H. C. W. X. X. C. Wang Xiulei, "SDICN: A software defined deployable framework of information centric networking," China Communications, vol. 13, no. 3, pp. 53 - 65, 2016.

[25] B. P. H. K. R. A. S. Samar Shailendra, "A novel overlay architecture for Information Centric Networking," in 2015 Twenty First National Conference on Communications (NCC), Mumbai, India, 2015.

[26] C. W. Abhishek Chanda, "A content management layer for software-defined information centric networks," in Proceedings of the 3rd ACM SIGCOMM workshop on Information-centric networking, Hong Kong, China, 2013.

[27] F. S. D. K. J. S. Markus Vahlenkamp, "Enabling ICN in IP Networks Using SDN," in 21st IEEE International Conference on Network Protocols (ICNP), Goettingen, Germany, 2013.

[28] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker and I. Stoica, "A data-oriented (and beyond) network architecture," in SIGCOMM '07 Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications, Kyoto, Japan, 2007.

[29] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs and R. L. Braynard, "Networking Named Content," in CoNEXT '09 Proceedings of the 5th international conference on Emerging networking experiments and technologies, Rome, Italy, 2009.

[30] C. Dannewitz, D. Kutscher, B. Ohlman, S. Farrell, B. Ahlgren and H. Karl, "Network of Information (NetInf) – An information-centric networking Architecture," Computer Communications, vol. 36, no. 7, pp. 721-735, 2013.

[31] N. Fotiou, P. Nikander, D. Trossen and G. C. Polyzos, "Developing Information Networking Further:From PSIRP to PURSUIT," in International ICST Conference on Broadband Communications, Networks, and Systems (BROADNETS), Athens, Greece, 2010.

[32] A. Detti, N. B. Melazzi, S. Salsano and M. Pomposini, "CONET: a content centric inter-networking architecture," in ICN '11 Proceedings of the ACM SIGCOMM workshop on Information-centric networking, Toronto, Ontario, Canada, 2011.

[33] M. Hoefling, F. Heimgaertner, M. Menth, K. V. Katsaros, P. Romano, L. Zanni and G. Kamel, "Enabling resilient smart grid communication over the information-centric C-DAX middleware," in 2015 International Conference and Workshops on Networked Systems (NetSys), Cottbus, Germany, 2015.

[34] F. Heimgaertner, M. Hoefling, B. Vieira, E. Poll and M. Menth, "A security architecture for the publish/subscribe C-DAX middleware," in 2015 IEEE International Conference on Communication Workshop (ICCW), London, UK, 2015.

[35] W. K. Chai, N. Wang, I. Psaras, G. Pavlou, C. Wang, G. G. d. Blas, F. J. Ramon-Salguero, L. Liang, S. Spirou, A. Beben and E. Hadjioannou, "CURLING: Content-Ubiquitous Resolution and Delivery Infrastructure for Next-Generation Services," IEEE Communications Magazine, vol. 49, no. 3, pp. 112-120, 2011.

[36] L. GuangChun, WangYanhua, L. Xianliang and Hanhong, "A Novel Web Application Frame Developed by MVC," Software Engineering Notes, vol. 28, no. 2, pp. 1-3, 2003.

[37] H. Derhamy, J. Eliasson, J. Delsing and P. Priller, "A Survey of Commercial Frameworks for the Internet of Things," in Conference on Emerging Technologies & Factory Automation (ETFA), Luxembourg, 2015.

[38] D. Happ, N. Karowski, T. Menzel and V. Handziski, "Meeting IoT platform requirements with open pub/sub solutions," Annals of Telecommunications, vol. 72, no. 1-2, p. 41–52, 2017.

[39] S. Kraijak and P. Tuwanut, "A survey on IoT architectures, protocols, applications, security, privacy, real-world implementation and future trends," in Wireless Communications, Networking and Mobile Computing (WiCOM 2015), Shanghai, 2015.

[40] "Fog Computing and the Internet of Things: Extend the Cloud to Where the Things Are," CISCO, 2015. [Online]. Available: https://www.cisco.com/c/dam/en_us/solutions/trends/iot/docs/computing-overview.pdf. [Accessed 10 1 2018].

[41] J. McKendrick, "Fog Computing: a New IoT Architecture?," rtinsights.com, 22 3 2016. [Online]. Available: https://www.rtinsights.com/what-is-fog-computing-open-consortium/. [Accessed 10 1 2018].