Software-Defined Networking: Current Trends, Challenges, and Future Directions

Abigail O. Jefia, Segun I. Popoola and Aderemi A. Atayero
Department of Electrical and Information Engineering
Covenant University
Ota, Nigeria
abby.jefia@yahoo.com
segun.popoola@covenantuniversity.edu.ng
atayero@covenantuniversity.edu.ng

Abstract

The ordeal which network operators face in implementing traditional network protocols has posed a great challenge to network management. As digital revolution of the world continues to transpire, demand has been placed on the exertion of high-level policies that will take network management to the next level. A new world of network programmability, software-defined networking (SDN), recommends the separation of the control plane and the data plane, enabling routers and switches to use information from the control plane to forward incoming traffic out the appropriate egress interface. SDN therefore provides a means for network virtualization. In this article, we present a review, which focuses on the concept of software-defined networking and the challenges for future networks. We also take a look at some issues currently being faced by SDN. In conclusion, a summary of the review is given, highlighting the need for SDN in a global world.

Keywords
Software-Defined Networking (SDN); network management; network programmability; Network Function Virtualization (NFV).

1. Introduction

The world is fast evolving and so are computer networks. These networks become more complex and as a result, managing them tends to place a serious challenge on the network operators. More devices are being added to the network infrastructure as time goes on and this facilitates the occurrence of various events in the network. The responsibility of configuring multiple "top-notch" policies to be administered on the network is placed on the network operator. Furthermore, the operator is faced with handling various network events, which may occur such as intrusions, etc. (Kim & Feamster, 2013). Today's networks continue to be vast and wide. A medium sized organization would contain hundreds or even thousands of devices connected to or in a network. In addition, Networks are heterogeneous (McKeown, 2009). Devices connected may be from different manufacturers, vendors and providers.

In order to enforce these "top-notch" policies, the network administrator would need to individually implement low-level and proprietary commands on each network device (Kreutz et al., 2015). Implementing these policies is therefore extremely challenging. These unending requirements lead to a paradigm shift in networking known as Software Defined Networking (SDN). SDN possesses several dynamic properties such as: scalability, adaptability, as well as its savvy design, which contribute greatly to its increasing global acceptance (Lali, Mustafa, Ahsan, Nawaz, & Aslam, 2017). It is currently a solution provider to conventional network problems and is gaining more acceptances in environments such as optical networks.

In this paper, we provide a short review on the concept of software defined networking. The review focuses on the performance evaluation of software-defined networking as against traditional networks and the challenges currently being faced by SDN. Section 2 explains the challenges of future traditional networks as network infrastructure becomes more dynamic and complex. Section 3 gives a definitive approach towards software-defined networking as well as its benefits. Section 4 presents the challenges currently faced by SDN. Section 5 deals with the different fields, which SDN solutions have been implemented. Section 6 concludes the paper.

2. Challenges of Traditional Networking in Future Communication Networks

Usage and demand for technology is growing at a very rapid pace. Networks continue to expand and become more complex as network infrastructure enlarges. In addition, new users and network services are constantly being added to the network. Both new users and network services demand huge network resources, which is increasing exponentially. With regards to increasing size and complexity of networks, traditional approaches for network management would be highly inefficient. This places a serious strain on network operators as they are faced with the task of implementing diverse configurations and keep track of innumerous events on the network.

There has been a massive increase in connected devices over time, going beyond what was assumed could be handled by data communication networks in the near future. For this operation, a solution to combat this substantial growth that can be maintained through a single point is essential. Such a solution must be able to provide services for futuristic needs and demands as well as providing ease of network management.

SDN is the key solution to the aforementioned problems. Through the concept of software-defined networking, network programmability is enhanced and network elements can be remotely managed from a centralized controller. The following sections discusses further on software-defined networking.

3. Concept of Software-Defined Networking

3.1. Definition

The Open Networking Foundation (ONF) has provided a clear-cut definition of SDN, which has become globally adopted (Parulkar, Sloane, Das, & Blair): Software-Defined Networking (SDN) is an emerging network architecture where network control is decoupled from forwarding and is directly programmable (McKeown, Anderson, Balakrishnan, Parulkar, & Peterson, 2012).

In a traditional router or switch architecture, both the control and data plane function on a single device. Software defined networking (SDN) is a network model that has been developed to virtualize the network. It functions by separating the control plane from the forwarding plane (data plane) in a network device. The SDN Controllers at the control plane perform all complex functions, as it is the “brains” of the device. On the other hand, the data plane is responsible for forwarding traffic flows (Pfaff et al., 2009). The controller is able to communicate with the switches through a communication protocol known as OpenFlow Protocol (Jarschel et al., 2011; Pfaff et al., 2009; Tootoonchian & Ganjali, 2010). Figure 1 illustrates the control and data plane structures in the traditional and SDN architecture while Figure 2 illustrates the SDN framework. In Figure 2, the use of Application Programming Interfaces (APIs) in the framework should be noted.

An API is a set of standardized requests, which define the proper means an application can use to request services from another application (Qin, Denker, Giannelli, Bellavista, & Venkatasubramanian, 2014). The SDN controller makes use of Southbound and Northbound APIs. The Northbound APIs are used to communicate with upstream applications while the Southbound APIs are used to define the behavior of the downstream virtual network devices (i.e. switches, routers, etc.).

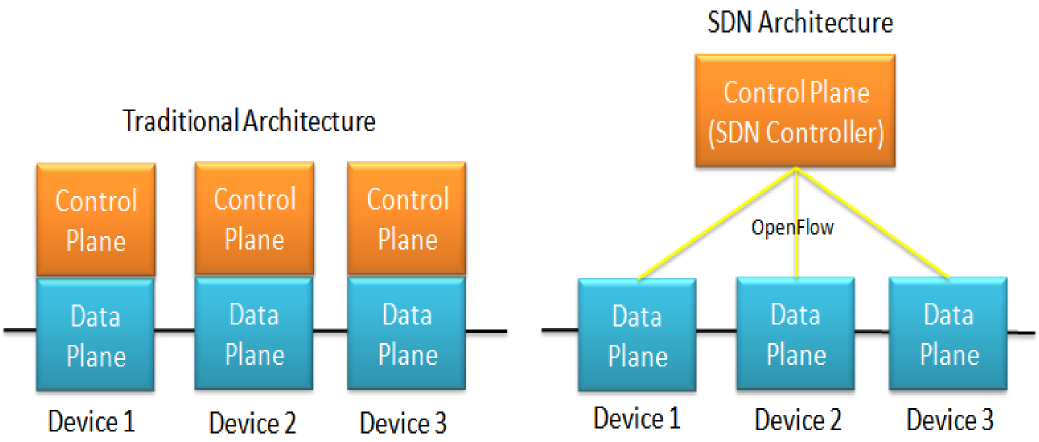

Figure 1. Traditional Architecture and SDN Architecture (Qin et al., 2014)

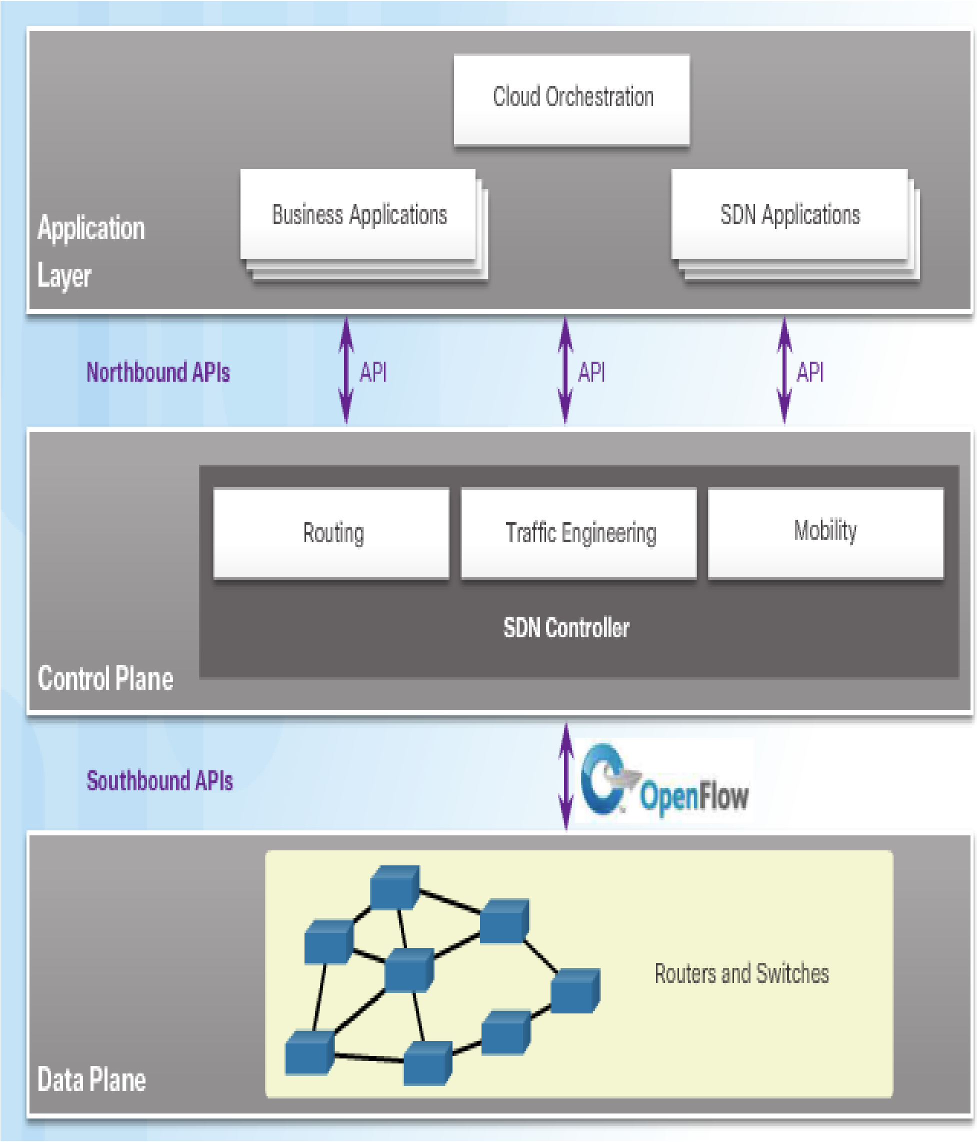

Figure 2. SDN Framework (Kreutz et al., 2015)

3.2. Benefits

Decoupling the control plane and the data plane provides better management of the network as well as flexibility by introducing programmability in the network. This structure enables network management to be carried out on the control plane, having no effect on data flows traversing the data plane. A few of the benefits of SDN include:

3.2.1. Enhanced Configurations

As network size continually expands and new devices are added to the network, proper configurations need to be implemented for effective network operation. Owing to the heterogeneous nature of network devices, a traditional approach is considered tiresome and fallible. With SDN, the numerous devices are connected to a centralized control plane where configuration and management is carried out from a single point.

3.2.2. Enhanced Performance

This is considered the main objective of SDN. Performance optimization is achievable through the centralized control plane, which SDN offers. As such, all challenges pertaining to performance optimization would become manageable. Consequently, traditional issues, which include; Quality of Service (QoS) support, end-to-end congestion, etc. can be developed and exploited to confirm their efficacy in enhancing system performance (Xia, Wen, Foh, Niyato, & Xie, 2015).

3.2.3. Reduced Cost

With SDN, the centralized control plane is responsible for managing and orchestrating the network, which can be handled by one single controller. In other words, SDN removes network infrastructure management and control away from the network devices and alternatively puts it into software, thereby reducing operating costs.

3.2.4. Encouraging Innovation

It is impossible to absolutely foresee and precisely meet the demands of future applications on networks, thereby leading to the deployment of applications, network services and new ideas. It provides a programmable network platform for the purpose of experimenting and carrying out implementations, which in turn enhances innovation.

4. Challenges of Software-Defined Networking

Although, SDN has been described as the key solution to the issues currently faced by the enlarging network infrastructure, it is still considered to be in its infancy stage. Benefits such as reduced cost, enhanced configuration, enhanced performance, alongside many others have been laid out, but various issues still require attention. Challenges continue to occur as SDN becomes widely adopted and new solutions are being proposed. In this section, we focus on key challenges posed by SDN, namely: (1) Scalability (2) Flexibility and Performance (3) Security and (4) Interoperability.

4.1. Scalability

This has been proven to be one of the major challenges posed by SDN. Two “sub-issues” can be derived from this single challenge namely: (1) controller scalability and, (2) Network-node scalability. A single controller can handle up to 6 million flows per second. Thus, this proves that either one controller or multiple controllers are able to administer control plane services required for a wide number of data forwarding nodes. To increase scalability, the logically-centralized controller must be physically distributed rather than operating on a peer-to-peer basis. However, the challenges faced by the controller as interaction occurs will be shared among network devices, be it a distributed or peer-to-peer controller infrastructure.

Onix and HyperFlow are considered to be effective approaches towards achieving scalability (Koponen et al., 2010; Tootoonchian & Ganjali, 2010). Onix operates by allotting and partitioning network state to various physically distributed controllers. HyperFlow is an application, which enables interconnection of independently managed

OpenFlow networks. The HyperFlow application specifically distributes events that make changes to the condition of the network, and the other controllers replay all the distributed events to reproduce the state. As such, every controller would function with the same homogeneous network topology. Yeganeh, Tootoonchian, and Ganjali (2013) discussed solutions to controller scalability as it relates to the provision of network view by distributing events over various controllers. Heller, Sherwood, and McKeown (2012) examined the issues concerning controller placement with respect to the number of controllers required and where they should be located in the network.

4.2. Flexibility and Performance

A basic challenge of SDN is the means by which to deal with high-level packet processing flows in a proficient way. Two key elements to be considered here are; flexibility and performance. In this section, flexibility refers to the capability possessed by networks to adjust to new and unprecedented features such as; applications and network services. Performance here deals with speed at which network nodes at the data plane process information from control plane.

It has been proven that general-purpose processors (CPUs/GPPs) provide the highest flexibility while Application-specific standard products (ASSPs) lay the foundation for high-performance networks. However, a hindrance to ASSPs is their limited flexibility. On another note, Application-specific integrated circuits (ASICs) are custom-built vendor-specific devices developed by organizations like Cisco and Juniper, that are put in use when the required standard products are unattainable and the programmable solutions are incompetent in meeting performance constraints. Nonetheless, ASICs provide the highest performance while offering lowest flexibility. The attributes possessed by ASICs provide the best requirement for implementing the SDN data plane and this is currently being incorporated into SDN products. Regarding the performance specification of data processing technologies, it can be deduced that a hybrid approach will provide and efficient and effective solution for SDN technology.

4.3. Security

Researches are being carried out till date concerning security problems associated with SDN. As SDN is widely being adopted and deployed, approaches towards security must be put into consideration. The Open Networking Foundation (ONF) had this in mind and they set up a security working group with them. Among various security vulnerabilities, authentication and authorization at the controller-application level are considered to be at the top of the list. In order to support network protection, an effective security model must be put in place.

In the SDN architecture, the controller is a target for threats especially when open to unauthorized access. Attacks on the controller can cause serious damage to the network, as it is responsible for controlling the entire network. Moreover, an attacker would be able to impersonate a controller and carry out malicious deeds. Transport Layer Security (TLS) is a security technology aimed at alleviating these threats due to authentication between controllers and their switches. TLS would provide the required security when implemented with a single controller, managing a group of network nodes. Nonetheless, authorization and authentication becomes more complex as a group of controllers interacts with a single node or vice versa. SDN architecture supports a high-level security system. It can support the following: Security policy alteration, Network forensics and Security service intrusion.

Various threat mitigation strategies would eventually arise even as the potential for unauthorized access increases. The best solution is for organizations to define an efficient high-level security policy to effectively attain network protection.

4.4. Interoperability

In this section, we consider how SDN solutions can be integrated into existing networks. We focus on the challenges faced in the migration from traditional to SDN approach. Deploying an entirely new infrastructure based on SDN technology would be adequate and for this to take place, all network elements and devices would be SDN-enabled (Sezer et al., 2013). Moreover, there is a tremendous base of networks supporting imperative systems and organizations, and transitioning these networks into new architecture is impossible as it is only targeted at infrastructure-based networks such as campus networks and data centers.

It can be deduced that transitioning to SDN requires the support of SDN and legacy equipment (Sezer et al., 2013). The IETF path computation element (PCE) would be able to assist in systematic migration to SDN (Paolucci, Cugini, Giorgetti, Sambo, & Castoldi, 2013). PCEP, which is a specific protocol facilitates the interaction between network elements. Although, PCE is incapable of providing complete SDN, the SDN controller is able to support complete path computation for the data flow across numerous network devices.

Researches and developments need to be implemented to attain a hybrid SDN architecture, enabling traditional, SDN-enabled, and hybrid network nodes to operate simultaneously. Therefore, in order to achieve interoperability, a protocol, which will offer compatibility with both the requirements for SDN communication interfaces, and existing traditional network protocols must be put to consideration. Numerous industry working groups such as the IETF, ONF, etc. are constantly proposing and developing standards and policies to facilitate migration from traditional to SDN model and their work must be harmonized.

5. Applications of Software-Defined Networking

Software-defined Networking can be implemented in various network environments. Owing to the partitioning of the control and forwarding planes, SDN enables customization as well as the deployment of new network services and policies. In this section, we consider various environments where SDN has been implemented.

5.1. Data Centers

The SDN architecture is targeted at Infrastructure-based networks and data centers fall under this category. As the world evolves and demand for high-level policies and network services emerge, data centers also make advances to meet these rapidly changing demands. Data centers operate large-scale networks, which places policy enforcement and traffic management at a critical level. These large-scale networks are subject to various challenges as they are often complex.

Abstractions have been proposed towards energy consumption in data centers. ElasticTree, proposed by Heller et al. (2010) is a network-wide power manager, which utilizes SDN to locate the areas of the network, which possess the minimum power requirement as well as meeting traffic flow pre-requisites and deactivates “non-useful” switches. Subsequently, they demonstrate energy savings under changing traffic conditions. One can only begin to imagine how much these savings can be increased if properly implemented with SDN.

5.2. Enterprise Networks

Generally, enterprises run large networks while possessing rigid security and performance requirements (Nunes, Mendonca, Nguyen, Obraczka, & Turletti, 2014). A campus network can be considered as an enterprise network where the connection of several temporary devices to the network is prevalent. This poses a challenge towards maintaining security and managing the network. For networks in an enterprise environment to function adequately, the SDN model can be adopted to as to enforce policies and services that would optimize network performance as well as enhance network management.

A study conducted by Reitblatt, Foster, Rexford, Schlesinger, and Walker (2012) presents challenges which arise from regular network updates. It was deduced that a typical and major source of network instability is consistent configuration change, which can lead to security limitations and performance disruptions. Furthermore, proposals have been made towards a group of high-level hypothesis that will enable network operators to make changes to an entire network, while ensuring that each packet crossing the network is handled by precisely one consistent global network configuration (Reitblatt et al., 2012). In order to support this concept, update mechanisms were developed in OpenFlow. With this, the challenges faced by enterprise networks will be addressed.

5.3. Optical Networks

Extending SDN to handle data traffic as flows enhances the integration of multiple network technologies. Consequently, communication between packet-switched and circuit-switched networks will be enhanced. The Open Network Foundation (ONF) had this in mind and created the Optical Transport Working Group (OTWG) in 2013 to provide the following benefits: conveying new administrations by utilizing virtualization and SDN, enhancing optical transport network control and management adaptability, and enabling the deployment of third-party administration and control systems (Pfaff et al., 2009).

Patel, Ji, and Wang (2013) presented a Software Defined Optical Network (SDON) architecture and a QoS-aware unified control protocol for optical burst switching in OpenFlow-based SDON is developed [20]. As a result, the SDON infrastructure supports unified control protocols that will optimize network performance and improve capacity (Nunes et al., 2014).

5.4. Home and Small Business

Although, SDN is targeted at large-scale networks, a number of researches have been made concerning its usefulness in smaller networks such as home or small businesses. Networks at this level utilize low-cost equipment, hence the need for tighter security and effective network management. Furthermore, it is impractical to have a dedicated network operator in every home and office. Calvert et al. (2011) explained that in order to manage home-based networks, the first step is to decipher what occurs in the network. They proposed implementing the network controller to act as a “Home Network Data Recorder” that would create logs to be used for troubleshooting or other functions. (Mortier et al., 2012) asserted that users are the ones who desire insight and management over the behavior of their network. They understood that most users are unable to implement traditional policies on networks and as such, they developed a prototype network in which SDN is used in providing a network-wide view to users as well as offering a single point of control.

6. Conclusion

In order to meet the demands of the increasing network infrastructure and advances in technology, SDN is considered a key solution to meet these demands. Initially, we reviewed the challenges awaiting future networks as complexity of the network infrastructure becomes more prominent. We drove at some points, which emphasize on the traditional approach to network management and how it would affect systems utilizing this mechanism in the near future. Furthermore, we provided an insight to the concept of the SDN architecture regarding its functionality, as well as highlighting its benefits such as enhanced configurations, enhanced performance, innovation, and reduction in cost.

Owing to continuous research being carried on the SDN model, we laid out some predominant issues affecting the efficacy and full utilization of the architecture with emphasis on scalability, flexibility and performance, security and, interoperability. It is believed that once these challenges are addressed and resolved, SDN would “sky-rocket” to the greatest height in technology advancement.

Finally, we examined various implementations of SDN on diverse environments such as in data centers, enterprise networks, optical networks and, homes and small businesses. The SDN model is still being tested and developed on numerous platforms to enable optimization of network management, which would lead to a software-defined revolution.

Acknowledgement

The authors wish to appreciate the Center for Research, Innovation, and Discovery (CU-CRID) of Covenant University, Ota, Nigeria, for the partial funding of this research.

References

Calvert, K. L., Edwards, W. K., Feamster, N., Grinter, R. E., Deng, Y., & Zhou, X. (2011). Instrumenting home networks. *ACM SIGCOMM Computer Communication Review, 41*(1), 84-89.

Heller, B., Seetharaman, S., Mahadevan, P., Yiakoumis, Y., Sharma, P., Banerjee, S., & McKeown, N. (2010). *Elastictree: Saving energy in data center networks.* Paper presented at the Nsdi.

Heller, B., Sherwood, R., & McKeown, N. (2012). *The controller placement problem.* Paper presented at the Proceedings of the first workshop on Hot topics in software defined networks.

Jarschel, M., Oechsner, S., Schlosser, D., Pries, R., Goll, S., & Tran-Gia, P. (2011). *Modeling and performance evaluation of an OpenFlow architecture.* Paper presented at the Proceedings of the 23rd international teletraffic congress.

Kim, H., & Feamster, N. (2013). Improving network management with software defined networking. *IEEE Communications Magazine, 51*(2), 114-119.

Koponen, T., Casado, M., Gude, N., Stribling, J., Poutievski, L., Zhu, M., . . . Hama, T. (2010). *Onix: A distributed control platform for large-scale production networks.* Paper presented at the OSDI.

Kreutz, D., Ramos, F. M., Verissimo, P. E., Rothenberg, C. E., Azodolmolky, S., & Uhlig, S. (2015). Software-defined networking: A comprehensive survey. *Proceedings of the IEEE, 103*(1), 14-76.

Lali, M., Mustafa, R., Ahsan, F., Nawaz, M., & Aslam, W. (2017). Performance Evaluation of Software Defined Networking vs. Traditional Networks. *The Nucleus, 54*(1), 16-22.

McKeown, N. (2009). Software-defined networking. *INFOCOM keynote talk, 17*(2), 30-32.

McKeown, N., Anderson, T., Balakrishnan, H., Parulkar, G., & Peterson, L. (2012). Software-defined networking: the new norm for network. *White Paper. ONF*.

Mortier, R., Rodden, T., Lodge, T., McAuley, D., Rotsos, C., Moore, A. W., . . . Sventek, J. (2012). *Control and understanding: Owning your home network.* Paper presented at the Communication Systems and Networks (COMSNETS), 2012 Fourth International Conference on.

Nunes, B. A. A., Mendonca, M., Nguyen, X.-N., Obraczka, K., & Turletti, T. (2014). A survey of software-defined networking: Past, present, and future of programmable networks. *IEEE Communications Surveys & Tutorials, 16*(3), 1617-1634.

Paolucci, F., Cugini, F., Giorgetti, A., Sambo, N., & Castoldi, P. (2013). A survey on the path computation element (PCE) architecture. *IEEE Communications Surveys & Tutorials, 15*(4), 1819-1841.

Parulkar, G., Sloane, T., Das, S., & Blair, C. Open Networking Foundation. *Open Networking Foundation*.

Patel, A. N., Ji, P. N., & Wang, T. (2013). *Qos-aware optical burst switching in openflow based software-defined optical networks.* Paper presented at the Optical Network Design and Modeling (ONDM), 2013 17th International Conference on.

Pfaff, B., Heller, B., Talayco, D., Erickson, D., Gibb, G., Appenzeller, G., . . . Casado, M. (2009). OpenFlow Switch Specification.

Qin, Z., Denker, G., Giannelli, C., Bellavista, P., & Venkatasubramanian, N. (2014). *A software defined networking architecture for the internet-of-things.* Paper presented at the Network Operations and Management Symposium (NOMS), 2014 IEEE.

Reitblatt, M., Foster, N., Rexford, J., Schlesinger, C., & Walker, D. (2012). Abstractions for network update. *ACM SIGCOMM Computer Communication Review, 42*(4), 323-334.

Sezer, S., Scott-Hayward, S., Chouhan, P. K., Fraser, B., Lake, D., Finnegan, J., . . . Rao, N. (2013). Are we ready for SDN? Implementation challenges for software-defined networks. *IEEE Communications Magazine, 51*(7), 36-43.

Tootoonchian, A., & Ganjali, Y. (2010). *Hyperflow: A distributed control plane for openflow.* Paper presented at the Proceedings of the 2010 internet network management conference on Research on enterprise networking.

Xia, W., Wen, Y., Foh, C. H., Niyato, D., & Xie, H. (2015). A survey on software-defined networking. *IEEE Communications Surveys & Tutorials, 17*(1), 27-51.

Yeganeh, S. H., Tootoonchian, A., & Ganjali, Y. (2013). On scalability of software-defined networking. *IEEE Communications Magazine, 51*(2), 136-141.

Biographies

Abigail O. Jefia is presently a senior undergraduate at Covenant University where she majors in Information and Communication Engineering in the Department of Electrical and Information Engineering with specialization in Networking. In 2015, she worked as an Information Technology Support intern at the Covenant University Network Office Centre where she developed her interest in computer networking. Afterwards, she interned with Telnet Nigeria Limited as a Network Analyst/Engineer in 2018, where she honed her network management skills and gained deeper insights into the various advancements towards networking. In 2018, Abigail obtained a Cisco Certified Network Associate Routing and Switching (CCNA Routing and Switching) Certificate from the experience garnered from her internships. Her current research interests include Software-defined Networking and Cyber security.

Segun I. Popoola completed his Master of Engineering (MEng) degree in Information and Communication Engineering at the Department of Electrical and Information Engineering, Covenant University, Ota, Nigeria with a

Distinction in June, 2018. He was the Overall Best Graduating Masters Student in Covenant University (2017/2018). Segun graduated from Ladoke Akintola University of Technology, Ogbomoso, Nigeria in 2014 with a BTech (*First Class*) degree in Electronic and Electrical Engineering. He was awarded the Best Graduating Student in the Department of Electronic and Electrical Engineering by the Faculty of Engineering and Technology in conjunction with the Nigerian Society of Engineers (NSE). He has authored and co-authored more than fifty (50) academic papers published in international peer-reviewed journals and conference proceedings. His research interests are, but not limited to: wireless communications, radio propagation modelling, Internet of Things (IoT), machine learning, and data analytics.

Aderemi A. Atayero is The Covenant University Professor of Communication Engineering and current Vice-Chancellor of the University. Professor Atayero has a Bachelor of Science Degree in Radio Engineering and a Master of Science Degree in Satellite Communication Systems in 1992 and 1994 respectively. He earned his PhD from the Moscow State Technical University of Civil Aviation (MSTUCA) in 2000. Atayero is a Fellow of the Science Association of Nigeria (FSAN) as well as a Senior Research Fellow of the International Association of Research Scholars and Administrators. Engineer Atayero is a COREN Registered Engineer and member of the Institute of Electrical and Electronic Engineers (IEEE) and other professional bodies. He has published over a hundred scientific papers in International peer-reviewed journals and proceedings. He is on the editorial board of several international scientific and engineering Journals. Atayero is a recipient of various awards and scholarships including the '2009 Ford Foundation Teaching Innovation Award'. His current research interests are in various aspects of Communication Engineering, including (but not limited to): Wireless Sensor Networks, Wireless (Mobile) Communications, Internet of Things (IoT), Smart Cities, and Cyber Physical Systems.