Master's in Telecommunications Program (ENTS)
SCHOLARLY PAPER SUBMISSION FORM

Students should submit this form with their Scholarly Paper by the stated deadline for graduation. Students should adhere to the guidelines of the Scholarly Paper Workshop, the ENTS Graduate Handbook, as well as the on-line Scholarly Paper Guidelines. Please, complete the following information:

Name:___Ademilola Aladetan____________________ UID: _______115800199____________________

Email:____adedavid@terpmail.umd.edu____________ phone: _____2404678688__________________

Graduation Term (ex: Fall 2017):______Fall 2019______________

Title of Scholarly Paper: Software-defined Networking: Business and Technical Analysis of a disruptive technology.

TERMS OF AGREEMENT AND SIGNATURE:
By signing below, I certify that that this Scholarly Paper does not reproduce material from a course, or overlap with any paper previously submitted for academic purposes, or plagiarize any anyone else's work. Furthermore, I pledge on my honor that I have not given or received any unauthorized assistance on this Scholarly Paper.
I understand that checking this box constitutes a legal signature confirming that I acknowledge and agree to the above pledge: ☒

Electronic Signature: Ademilola Date: 10/11/2019
(Type your name in the box)

For Program Office use only (do not type on this page below this line)

First Submission: ☐ Approved ☐ Denied Initials: _________ Date: ___/____/____

Comments: ____

____

____

____

____

Second Submission: ☐ Approved ☐ Denied Initials: _______ Date: ___/____/____

Comments: ____

____

____

____

____

Software-defined Networking: Business and Technical Analysis of a disruptive technology

Ademilola Aladetan
115800199
11th October, 2019

I pledge on my honor that I have not given or received any unauthorized assistance on this assignment, that my paper contains no plagiarized material, and that I have made every effort to cite sources properly.

10/11/2019

Student Signature Date

Abstract

What drives competition and makes a market lively and developed is Innovation. Innovation is central to the survival of any firm. What then is innovation ? It is the process of creating a new product that customers perceive to be novel. Innovation can be incremental or breakthrough (or disruptive as coined in some literatures). This paper views Software-defined Networking, an emerging technology in the IT industry as a disruptive innovation. SDN is a novel Network architectural framework that enables networks to be programmed and centrally controlled to support business objectives. Big players in the industry initially downplayed the impact of this technology, describing it as a media hype. With widespread support from the Academic, research, and Open source communities, SDN gained significant adoption in data center, Enterprise and Service provider networks, bring along in its trail, a new set of players to compete with leading players like Cisco and Juniper. Though on a long recovery trajectory, Cisco systems appears to be the most impacted Network equipment manufacturer. Cisco is a leading member and current supporter of many SDN-affiliated bodies, but this paper believes the firm acted too late in aligning with the SDN movement.

I. Introduction

It's generally believed that the theory of **"disruptive innovation''** was coined by professor Clay Christensen who is the Kim B. Clark **Professor** of Business Administration at the Harvard Business School [9] . According to the theory, established or successful market leaders who fail to innovate run the risk of being overtaken by revolutionary newcomers. The activities of these newcomers have the potentials of creating a new market, replacing an existing product or making its industry obsolete. Disruptive players usually enter a market, starting off with low performance as niche players to meet some specific needs that are left unaddressed by established market leaders. Such is the condition of the Computer Networking industry, which has stagnated for decades due to lack of a competitive landscape.

However, the Networking market is currently witnessing an innovation-driven growth as a result of the adoption of SDN. What exactly is Software-defined Networking ? The definitions of SDN are varied and sometimes confusing. In some cases, it's difficult to know which SDN implementations conform to the original definition coined by the progenitors of this paradigm-changing technology. In a general term, SDN is a framework where the control planes (intelligence) of network devices are decoupled from the devices' data plane (hardware elements responsible for packet forwarding) and centrally implemented in a software-based Controller. The Controller that has the ability to program the data elements and presents an abstraction of the underlying infrastructures (physical or virtual) to higher-level applications such as Management, cloud orchestration or User-defined applications that can dynamically control the entire network infrastructure based on business rules or application requirements. A typical SDN framework has three broad components: SDN Controller, SDN Applications and the underlying Network elements (known as SDN devices).[6]

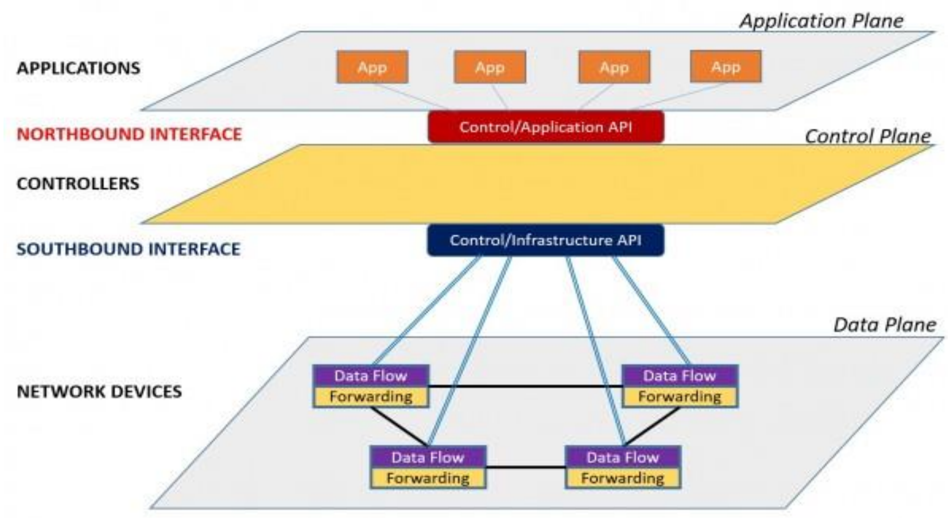

Figure 1 - Software-Defined Networking – A high level architecture

Source[6]

From figure 1, the bottom of SDN architecture is made up of SDN devices, which are usually physical or virtual switches. They are responsible for forwarding, dropping or modifying packets within a network based and based on the entries in their flow tables. Flow tables are forwarding logic that defines what action must be taken on traffic flows. A traffic flow is defined as a set of packets that share common characteristics and must be given the same treatment by the network device. such characteristics include destination subnet, VLAN, port, QoS etc.

The idea of having network devices becoming programmable by upper-layer applications was greeted with cold feet by established players who feared that their revenues would be negatively impacted. In fact, Cisco once played down the significance of SDN as it made frantic efforts to defend its market position and for fear that its gold-mine (routing and switching hardware) may end up becoming mere commodity devices. But as the adoption of SDN gains significant momentum, we're beginning to see a lot of acquisitions, strategic alliances, startups across all layers of the stack, new Vendor ecosystem in today's Networking market as both the old and new players seek to gain market share and consolidate their position. The inability of a leading player like Cisco Systems to respond early to unfolding trends in the market, according to industry experts, is the reason its market capitalization plummeted by also 50% over a period of 10 years[7]. The business analysis section of this review focuses mainly on Cisco's response to the disruptive activities of SDN in an industry where it once reigned like a demi-god.

This paper is organized as follows: Section I will review the limitations of legacy networking architecture and its inability to support emerging business objectives, Section II will discuss the meaning, genesis and different implementations of SDN, Section III will introduce us to the newly formed SDN market, the SWOT analysis and corporate strategies of Cisco Systems as it seeks to recover from a massive market share loss.

II. Emerging Business needs and the limitations of legacy Networks.

Legacy computer networks are usually implemented with routers, switches, and a couple of other middleware devices like firewall, load balancers and Cache engines. These devices were designed to have embedded intelligence and to operate in a distributed fashion. Each device has a control plane where control functions like exchanging of protocol information with other devices in a network-wide collaboration are implemented. Decisions regarding routing of incoming packets, preferential treatment of packets that require quality of service, or dropping of packets defined in access control lists are taken in the control plane but implemented in the data plane of the device.

As Data Networks became increasingly complex and large, and due to advancement in the design of powerful application-specific integrated circuits, functions like QoS, ACLs, switching, layer 3 forwarding were moved to the hardware (low level functions). Thus, network devices like Carrier-grade routers and Multi-layer switches were able to treat and forward packets at wire-speed such as 10Gbit/s, 100Gbit/s and 400Gbit/s[2]. This incremental innovation is good, but the underlying problems remain. Network devices must be operated hop by hop. This makes configuration of such devices to be prone to errors and slow to implement, especially when dealing with a large Network. Another problem with legacy Network is management. Managing a multi-vendor Network in a hop by hop fashion from the command line interface (CLI) or SNMP could be too complex and frustrating to Net Admins, just to say the least. Each node was designed to make a forwarding decision. The complexity of forwarding decisions increases with the number of nodes in the network. Net Admins must learn how to use vendor-specific CLI commands and Network Management systems to be able to support their Networks. Businesses were required to employ and train large number of Net Admins and Engineers for them to cope with the daily task of Network management. Changes to the Network must be scheduled weeks or days ahead of time. Approval for those changes are sometimes declined due to the fears of the unknown. Thus, business operations suffered due to inefficiencies inherent in legacy network architectures.

Furthermore, due to vendor-lock in, legacy networks have remained static and closed to innovations for decades. Network equipment manufacturers used vendor-specific and proprietary software in a fashion that stifled innovations and prevented businesses from making Networks to adapt to changing application needs. IT operations have witnessed tremendous efficiency in resource utilizations and performance due to virtualization technologies, but such gains were limited to Server and Storage resources while the Network remained a bottleneck. The huge benefits of open source software from the various communities could not trickle down to the Network resources because the devices used for implementing Networks remained under “lock and key”.

Finally, the increasing cost of implementing and maintaining legacy Networks has proved prohibitive for many businesses. Industry giants charged a lot of money for their Networking devices. Margins from these devices were as high as 30 percent despite the fact that many of them were sold with base features. Feature upgrades attracted huge cost in license fees. Apart from huge Capital expenditures, businesses were made to part with enormous OPEX like costs for technical support, large internal Network team and various kinds of Network management software. These expenses can be enormous if the Network infrastructure is multi-vendor and large.

It has become obvious from the foregoing that legacy networks could not support many business objectives. In a bid to overcome a lot of these challenges, many businesses started migrating their applications to the cloud. Cloud computing, which is the on-demand delivery of hosted services or resources like compute power, storage, databases, data analytics, software applications on a payment term that is generally described as ‘pay-as-use’, has become the life saver of many businesses. The benefits of Cloud computing range from speed of deployment, cost savings, global scalability, resource elasticity to service reliability and agility. The increasing adoption of the Cloud as a new business model boils down to the fact that both small and large businesses are now realizing that business agility is a differentiating factor that can put them ahead of their competitors. Being able to deliver services faster and adapt to changing market conditions is how to win in the market and that is what business agility is all about. Cloud computing offers this benefit, in addition to cost efficiencies, efficient and dynamic allocation of resources and the use of Big Data analytics for business intelligence. To cope with influx of users' application and data into the Cloud, Cloud providers were compelled to build large and several data centers at designated points around the globe[4].

These data centers were built with virtualization technologies (server and storage) and over time, their complexities and the need to support changing application needs proved to be beyond the capabilities of legacy network architecture. The desired outcome of this situation is to have network infrastructure as a code by a way of network programmability. The only way this could happen is to virtualize core network functions and have them implemented in software. For this model to be realized, Networking devices must have their control functions separated from the individual boxes and implemented centrally in a software-based Controller where upper-layer Applications can programmatically control them via APIs. The underlying networking devices can be thought of as mere commodity devices where actual packet forwarding is taking place. This is model is called software-defined Networking. Data centers thus became the main driver behind SDN.

The other use case where SDN has become very attractive is in the area of Wide Area Networking. The is generally known as SDN-WAN. The technology is manly applied in Enterprise Networks where it's used as a centralized policy-based Controller to provide an abstraction of management and monitoring functions at the edge of Enterprise Networks. Examples of WAN connections where it is being applied include the internet, 4G, LTE, MPLS and T1 circuits. This technology enables Enterprises to adopt cost effective and flexible WAN solutions without using dedicated proprietary devices.[4]

III. Genesis, Characteristics, and various Implementations of SDN:

Several attempts aimed at re-defining Networking architecture to achieve operational and cost efficiencies were made in the late 90s and early 20s. One of these efforts, originating from the IETF standard body in 2003 was called Forwarding and Control Element Separation (ForCES). The implementation of this standard requires clear separation between the control plane (Control Element) and forwarding plane (Forwarding Element) of a network device (called Network Element). The Control element may be implemented either in the same network element or in a separate node called the Controller. Communication between the CE and the FE is via a protocol called ForCES protocol. The strength of ForCES lies in the fact that capability exchange is usually negotiated between the CE and the FE. Also, legacy routing protocols can be used for communication between CE and FE. Though not a popularly adopted framework, the communication protocol of ForCES became the foundation of the Ethane project in 2007.[1]

Most of the concepts that formed the building blocks of today's SDN were earlier conceived by some researchers at Stanford University in a project was called 'Ethane'. By making flow entries in the forwarding tables of underlying networking elements, the project explained how high-level policies can be used to influence traffic or packets flow through a network in a programmatic fashion. To follow through on this project, the researchers decided to implement the project on Stanford Campus Network using locally developed Layer 2 Switches known as 'Ethane switches'. A key component of the project was the development of communication interface between

Network controller and some bare metal Switches whose flow tables could be manipulated programmatically. This effort inspired the creation of OpenFlow in 2009.
As OpenFlow began to gain acceptance in the Networking community and among researchers , the Open Networking Foundation (ONF), a growing community of operators, vendors and System integrators took over the promotion, standardization and use of OpenFlow protocol as its de facto platform-agnostic standard of communication between any Open SDN controllers and Network devices (physical and virtual devices) in the southbound direction. SDN Controllers have northbound interface which connect to cloud orchestration, management systems but there is no standard protocol defined for communication. REST or Java/Python APIs have used successfully.[5] With the acceptance of OpenFlow as a standard communication protocol for SDN controllers, many networking vendors started producing OpenFlow enabled switches. New startups like Big Switch Networks, Plexxi, Pica8, Nicira made their foray into the new SDN market. Industry big players like Cisco, Juniper, Huawei, NEC , all announced their support for SDN. Thus, around 2011, a new market of SDN products and services has its stage set up for fierce competition, mostly between the incumbents and the newcomers.

it's important to know that SDN is a framework, not a protocol. The interesting thing about this framework is that additional features to the framework usually come in the form of plugins. With several flavors of SDN in the market, it's necessary to define the features of a real Open SDN. A true open SDN is known to five fundamental traits. These include the followings[1]:

(1) plane separation: In a true and Open SDN implementation, control plane is completely separated from the data plane. Both planes are decoupled but communicate via a standard protocol as shown in Figure 2 below.
(2) Simplified device: Another feature of an Open SDN is that the data plane, which forwards packet flows based on the actions defined in its flow table, is a simple, commodity, less costly device whose sole function is packet switching. It can be a regular network device or any of the x86-based server with some special application-specific integrated circuits designed for packet forwarding.
(3) centralized control: Any SDN implementation that still permits the underlying SDN devices to perform control functions is not an Open SDN. The goal for centralizing control function is to have a single point of contact between policy defined in the upper layer applications and the underlying network devices in a network-wide operation.
(4) Openness: Openness means both the southbound and northbound interfaces used by SDN controllers are standard non-proprietary communication protocols. This unique feature is what promotes innovation, enhances interoperability of devices from different vendors and drives down the cost of network devices. Two Open southbound protocols defined for SDN controller communication are OpenFlow and OVSDB (Open vSwitch Database Management protocols. For SDN controllers' northbound communication, some popular APIs written in REST or Python seem to be sufficient at the moment.
(5) Network automation and virtualization: SDN enables Network services or functions to be abstracted from networking devices such that user-defined applications can define logic for how these services should behave. This is called Network function virtualization and it is the foundation of network automation. Any SDN implementation that does not fulfill this requirement cannot be described as Open.[5]

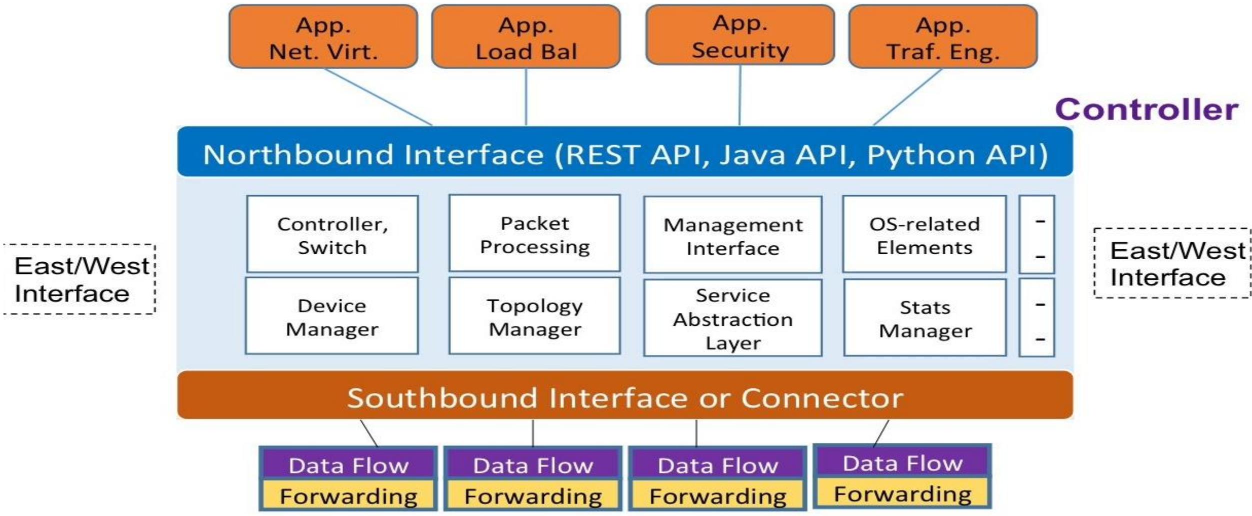

Figure 2: Open SDN Architecture. **Source[6]**

A number of commercial SDN Controllers from leading Network Equipment Manufacturers like Cisco, Juniper and Arista support proprietary APIs, in addition to OpenFlow, for backward compatibility with legacy equipment. For example, OpFlex protocol is used on Cisco's APIC SDN Controller to communicate policy definitions to underlying SDN devices, which still maintain their intelligence and run control functions like routing protocols. SDN Applications at the upper-later mostly use REST APIs to communicate with the SDN Controllers on the northbound interface. The SDN controllers in turn communicate with Network elements using either NETCONF API, Border Gateway Protocol Link State (BGP-LS), Path Computation Element Protocol (PCE-P) or the OpenFlow protocol. A lot of new Networking gears from these Vendors now support YANG data motel . The YANG data model is an initiative of the IETF to help vendors in the Networking industry achieve a common definition of configuration and state data structures of network devices. NETCONF on the other, is a protocol that uses SSH to communicate XML formatted YANG data with network devices[1]. This implementation is known as SDN via APIs
Three broad implementations of SDN have been identified so far. The first and second variants , known as Open SDN and SDN via APIs, have been described in the last two paragraphs.

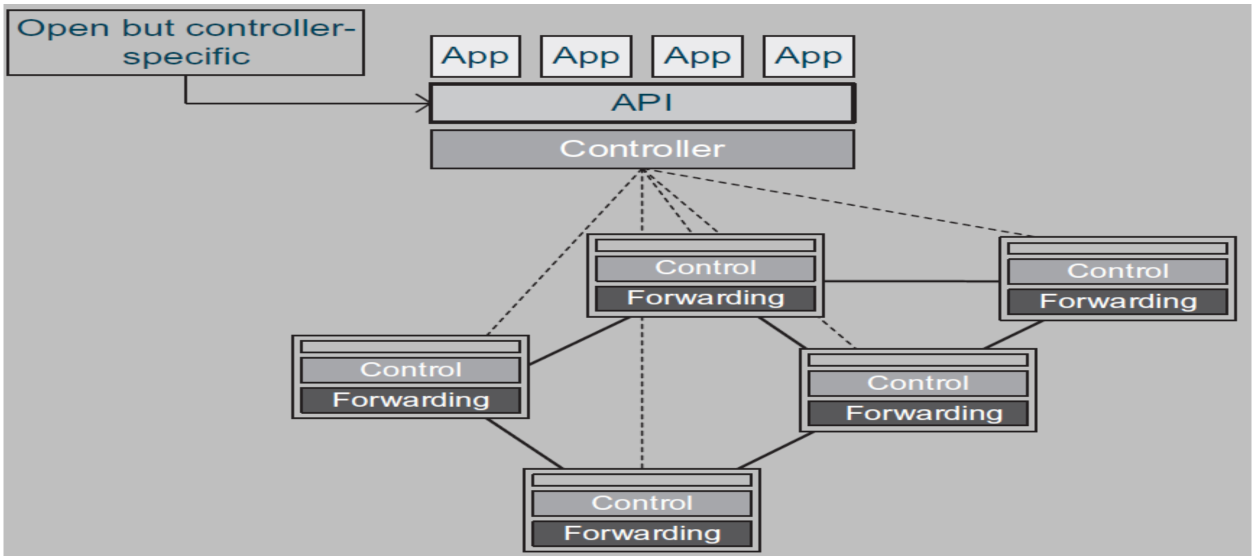

Figure 3: API-based SDN Architecture. **Source[1]**

SDN via APIs remains a good choice for hybrid networks that must support both SDN devices (using OpenFlow) and legacy devices (NETCONF, PCE-P, BGP-LS).
In the third variant of SDN implementation, Hypervisors are used to create virtualized L2 networks on top of physical network infrastructure. Virtual switches, positioned at the edge of virtual networks, are called virtual tunnel endpoints (VTEP). Two VTEPs communicate with each other via tunneling protocol such as Virtual Extensible Local Area Network (VXLAN), which is the most popular of all the protocols used for implementing overlay Networks. Each virtual switch has a list of other virtual switches and all MAC addresses reachable through those virtual switches. frames sent to a virtual switch is encapsulated in a 24-bit VXLAN header and forwarded to a remote switch where it is de-encapsulated[2]. The protocol is backed by VMware and Cisco. VMware's NSX virtual platform is based on VXLAN. Other tunneling protocols competing with the VXLAN in this space are Network Virtualization Using Generic Routing Encapsulation (a protocol backed by Microsoft, Dell, HP and Intel), and Stateless Transport Tunneling Protocol (whose draft is currently with the IETF).[8]
It must be noted that SDN via hypervisor-based overly network does not address a lot of the fundamental problems seen with legacy networks but rather presented a solution that delivered on Network automation and virtualization, finding applications mainly in the data center networks.
Looking across all the three variants of SDN implementation, several groups of players that make up the SDN ecosystem (commercial groups only) can be categorized as shown below:

1) Leading Network equipment manufacturers, e.g. Cisco, Juniper, IBM, HP, Arista, Extreme, Brocade, NEC. Most of these players are leading members of OpenDaylight project, a body devoted to promoting Open SDN and Network Function Virtualization. The project is being hosted by the Linux Foundation. ODL Controller is one of the most popular Open SDN controllers available today. The commercial versions of the ODL SDN controller from the Vendors identified above support both OpenFlow protocol and proprietary APIs.[1]]
2) Software vendors, e.g. VMware, Microsoft, Big Switch. VMware and Microsoft offer SDN products that are based on Overlay model. Big Switch, a startup in this space, produces Network operating system, known as Switch Light and maintains alliances with white-

switch manufacturers to offer OpenFlow based SDN products. Other startups that produce Software that can be used on white-box Switches are Cumulus and Pica8.

Others are the silicon chip makers (Broadcom, Intel who produce mainly switching chips for bare-metal x86-based hardware) and the white-box Switch manufacturers. White-box Switches are high-performance open Switches that are capable of running any network operating systems. They are built in a fashion similar to x86-based servers that can be installed with any Operating system. White box Switches can be purchased either as bare-metals commodity hardware or preloaded with a Network OS. The main attraction of these devices is their low cost.

IV. Cisco's external competitive environment and corporate strategies for recovery

The greatest competition in the SDN market is between leading Network equipment manufacturers and Software vendors who form alliance with white-box Switch Manufacturers to offer low cost alternative networking gear. The rest of this analysis will focus on the business strategies of a leading SDN player like Cisco. The analysis will review Cisco's internal and external environments, strategic alliances, acquisitions, diversification and product differentiation.
Cisco Systems has dominated the Networking industry for decades. In 2000, it ranked third behind Microsoft and General Electric in total market capitalization, having surpassed a $450 Billion mark[4]. Across all Networking sectors, Cisco boasted 54% of total market share in the early 20s. The figure 4 below shows Cisco's Market Capitalization values from Year 2006 – 2018. Worst performances were recorded from 2007 – 2009, the years leading to the adoption of OpenFlow as standard protocol for SDN solutions.

Fig 4: Cisco Market Cap 2006-2019 | CSCO. Source[7]

Any firm seeking to gain a competitive advantage over its competitors must adopt some strategic choices based on the outcome of the analysis of both its external and internal environments. This strategic tool is called SWOT analysis. External analysis will focus on both the threats and opportunities that a firm is faced with while internal analysis is concerned about revealing the strengths and weaknesses in both the resources and capabilities of a firm. Competitive advantage can be for short term. If it must extend into the long term, then it must possess qualities like being valuable, rare and imitable. In analyzing its external environments, a firm seeks to identity threats coming from the following sources: 1) New competition, 2) existing competitors, 3) substitute products, 4) influence of buyers.[3]

The decline in Cisco's market share and capitalization is attributable to three of the four external threats identified above. These threats are

(i) influence of buyers: The world's biggest buyers of white-box switches, Facebook, Amazon and Google, were once powerful and influential buyers of Cisco networking gears but that is no longer the case. Also, Verizon and AT&T who were once heavy consumers of Cisco devices, have also gone the path of white-box switches. These buyers are leading Big Techs whose actions have the capacity of influencing the direction of things in the industry. Facebook's announcement of its Open Compute Project in 2011 finally put Cisco on notice.

(ii) New competition: low barriers of entry from SDN has opened the Networking space to newcomers like Big Switch Networks, Cumulus networks and Pica8. Established players like Microsoft and VMware were not previously competing in the Networking space but have also moved in to carve a niche for themselves.

(iii) Availability of substitute products. Cisco and other established NEMs made fortunes from the sales of their networking gears before the boom of SDN. Their products were preloaded with Control and Management software bundled together. Software upgrade and technical support attracted huge amount of money. But that has changed, all thanks to SDN. The advent of low-cost white box switches preloaded with Network Operating System is a great substitute that enterprise businesses have happily embraced.

Apart from analyzing its external threats, A firm in any competitive industry must be able to identify opportunities that are unfolding in its industry and seize on them. Cisco Systems appears to have done exactly that through its support for many of the ongoing open source initiatives and projects. It's a leading contributor to the OpenDaylight project. Many of Cisco's new products are SDN compatible. A number of recent acquisitions and alliance formations are strategic moves aimed at repositioning Cisco in the evolving SDN market. The figure 3 above shows that Cisco market capitalization has been on the rise from 2012 till date.

A review of Cisco's internal environments will reveal some notable strengths that the firm is noted for. They include strong channel structure, large market share, good product design, a strong R&D and an impressive technical knowhow. However, like every other firm, Cisco has certain weaknesses that appear to have impacted its image and performance. These include high cost of products and support, lack of custom-tailored products for small buyers of networking devices, and slow response to major market trends. However, events in the last six years attest to Cisco's renewed commitment to regain a leading position in the Networking market through some deliberate corporate-level strategies. We shall now focus on what these strategies are.

Product differentiation: differentiated products are products with better perceived value. There are two approaches to achieve this objective. A firm can achieve product differentiation by in the area of superior product attributes or an unmatched competence in customer relationships[3]. As a leading member of the Opendaylight project, Cisco offers a commercial version of ODL SDN Controller. In addition to this, it also offers other non Open SDN Controller like APIC-DC and other SDN devices accessible via proprietary APIs as a differentiation strategy.

Diversification: it's a corporate-level strategy where a firm decides to operate in several industries (product diversification). The end goal for diversification is to produce economies of scope that either reduces cost or increases revenues or achieves both business objectives[3].
As businesses shift away from on-premise IT model, cloud-based products and services are seeing more adoption. The issue of security is getting more focused attention than ever. Cisco has aligned itself with this reality by diversifying into software, security and other subscription-based services. HyperFlex is a good example of Cisco's pay-as-you-go software offering that integrates networking, storage and computing resources in a converged system. Cisco's acquisition of Springpath made this diversification easy and strategic. Market analysts have attributed Cisco's steady rise in revenue to this effort.

Strategic alliance: strategic alliance is when two or more firms mutually agree to develop or market a product. Motivations for such moves include the desire to avoid the high costs required to operate in a new market, neutralize threats from competition, manage uncertainty and risks. Cisco's alliance with F5, a firm that specializes in application services that run in private, public and multi-cloud environment, enables Cisco's APIC Controller to be used as part of Cloud security platform. Another important alliance involving Cisco, EMC and VMware was consummated in 2019 as Cisco seeks to deepen its presence in the hyperconverged infrastructure industry.

Acquisitions: is a form of vertical integration where a firm acquires another firm for reasons ranging from desire to increase market share, gain better competitive advantage or penetrate new market[3]. It's also a fact that some firms acquire other firms in order to ward off competition posed by the acquired firms. Cisco was considered the largest acquirer of SDN-related companies in the networking industry. Table 5 below summarizes Cisco's acquisitions of SDN-related firms and the strategic reasons for such purchase, from 2012 till 2019.

Firm acquired	Year	Desciption	Strategic reasons
Cariden	2012	The core software in Cisco's WAN Automation Engine	To further consolidate the gains of Cisco's APIC Controller since WAE is used as SDN application for IP/MPLS automation
Meraki	2012	Cloud-based Software to manage, provision and configure wireless Access Points and wired switches.	Meraki remains Cisco's major Cloud-based solutions for Enterprise market.
Insieme	2013	Data center or cloud solutions used for the management of physical and	To increase market share in the SDN market

		virtual devices. The platform is called Application-centric Infrastructure (ACI)	
Tail-f	2014	Software application is used as orchestration tool to permit the management of both old OSS and SDN-based OSS systems in Service provider networks	Cisco has had tough time penetrating service Provider's market. This acquisition was made to increase Cisco's share in that market.
Embrane	2015	Software is used to create application-centric network services like firewall, load balancers etc.	The core offering of Cisco's Network Function Virtualization product. It helps Cisco to further gain weight in the SDN market.
Viptela	2017	SDN-WAN Solution for Enterprise	To extend the reach of its SD-WAN product into the Cloud.

Figure 5: List of Cisco's acquisition of SDN-related firms

II. Conclusion

There is no doubt that SDN is a solution to many long-standing networking problems such as rigidity, lack of efficient resource allocation, high cost, vendor lock-in, complexity and stifled innovation. Its adoption will be extended to more use cases as the technology matures. As usual for any new technology, bugs in SDN features will continue to be tolerated during the introduction and growth phase of its product life cycle. Cisco can strategically take the lead in defining standard and working closely with the Open source communities to further enhance the capabilities of Software-define Networking. When SDN finally moves to its maturity stage, its user base will be much larger than it is currently.

IV. References

[1] Paul Göransson et al. Software Defined Networks A Comprehensive Approach. Cambridge, MA: Morgan Kaufmann, 2017, pp. 1-87, 253-268, 327-372.

[2] Patricia A. Morreale et al. Software Defined Networking: Design and Deployment. Boca Raton, FL: CRC Press, 2015, pp. 51-78.

[3] Jay B. Barney and William S. Hesterly. Strategic Management and Competitive Advantage. Upper Saddle River, NJ: Pearson Education Inc., 2012, pp. 28-64, 248-276.

[4] Jim Doherty SDN and NFV Simplified. LSC Communications, VA: Pearson Education, 2016, pp 119-130, 170-190.

[5] Thomas D. Nadeau and Ken Gray. Software Defined Networks. 1st Edition, 1005 Gravenstein Highway North, Sebastopol, CA: O'Reilly Media. 2013, pp 71-100

[6] : Doan B. Hoang, Software Defined Networking ? Shaping up for the next disruptive step? Journal of Telecommunications and the Digital Economy, Vol 3, Num 4, Article 28, Nov. 2015

[7] Macrotrends, Stocks and Market capitalization reports :
https://www.macrotrends.net/stocks/charts/CSCO/cisco/market-cap

[8] Zhang Y, Natarajan S, Huang X, Beheshti N, Manghirmalani R. A compressive method for maintaining forwarding states in SDN controller. In: Proceedings of the third workshop on hot topics in software defined networking, HotSDN’14. New York, NY, USA: ACM; 2014. p. 139–44. http://dx.doi.org/10.1145/2620728.2620759.

[9] Clayton M. Christensen, et al. What Is Disruptive Innovation? Harvard Business review, Dec. 2015 issue, pp 44 - 53