

## Software Defined Networking (SDN) Research Review

Zhao Zhang<sup>a</sup>, Hailong Li, Siqi Dong and Lei Hu

Rocket Force Army of Engineering University, Shanxi Xi'an 710025 China;

<sup>a</sup>657455664@qq.com

**Keywords:** Numerical control separation; Flow scheduling; Controller

**Abstract.** Web Abstraction Leads to the Generation of Software-Defined Networking (SDN). SDN unique unique CNC separation structure to control the liberation. The development background of SDN is introduced, the most mainstream SDN structure is analyzed, which is divided into three layers: application plane, data plane and control plane. The key technologies of each plane of SDN three-layer structure are studied, and the advantages and disadvantages of each technology are compared. Then the existing application fields of SDN were investigated, and finally the development trend of SDN was prospected.

### 1. Introduction

With the rapid development of network technology and the increasingly complicated interlacement between human life and network, the scope of the Internet is getting larger and larger. The traditional network architecture and the carrying capacity of devices to the network are undergoing tremendous tests. Software defined network (SDN) separates control functionality from traditional networks to create a centralized logical control plane that can be used to formulate services and rules for different needs at the root, and to translate these into data- Flow table issued. This mode of operation provides greater space for increased network utilization and cost savings.

In the traditional routing mode, both the inside and outside of the domain are forwarded according to the destination IP. The control plane temporarily stores the routing information in the routing table and stores the IP and other information in the FIB of the TCAM during forwarding. When faced with large-scale network traffic, routing forwarding work faces enormous challenges.

It is the traditional network architecture, there are a variety of problems, new, revolutionary network technology and architecture almost certain. SDN technology, which originated from Stanford's Clean Slate research project [1] in 2006, introduced the idea of CNC separation in 2009, separating control from data forwarding. The data layer switch only bears the forwarding function, and the control plane will be all the information and control functions delivered to the controller unified deployment. The use of a unified interface between two planes for information interaction. The controller sends the forwarding rules to the lower switch through the interface protocol, and the switch only needs to be executed according to the rules. This improves the overall control performance and reduces the cost.

The successful case of SDN technology deployed by Google in 2012 on the backbone network has increased the utilization of the network to more than 70% and has become a landmark event in the application of SDN technology. Since then, the world's major companies have subsequently released their own solutions for SDN thereafter. SDN appeared at Ethane [2], a Stanford University laboratory research project, but has become a hot spot in the world for network technology and is considered as one of the key technologies of the next generation Internet [3].

This article starts from the three aspects of the most mainstream architecture of SDN and focuses on the tasks and main functions of the three planes in the whole system. Then, according to the data and control plane, the key technical trends of the current research are summarized. At the same time, the application scope of SDN is analyzed. Finally, the prospect of SDN in the future industrial application is prospected.

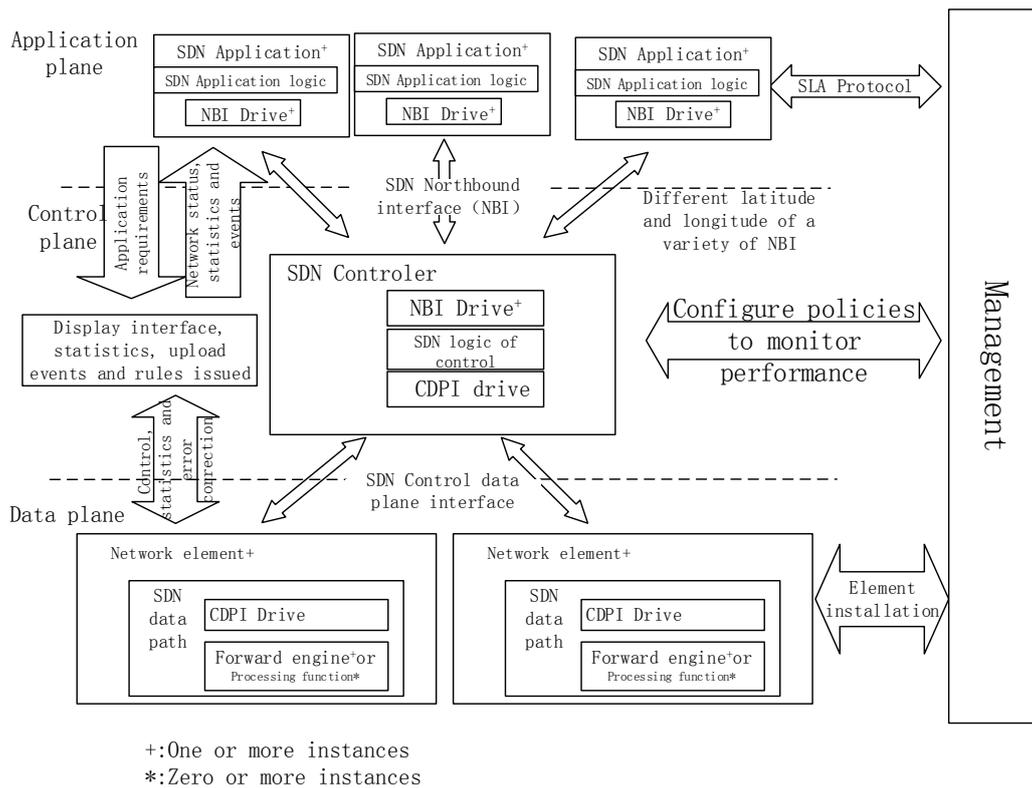


Fig. 1 SDN architecture

## 2. SDN Architecture Introduction

### 2.1 SDN Development Background.

The main idea of SDN architecture is control and data separation, that is, control logic centralized and data forwarding separation. This idea originated in the proposed OpenFlow protocol. In 2006, the United States National Science Fund at Stanford University (NSF) and seven other companies opened the Clean Slate research project and proposed the Ethane network architecture in the project. In this network architecture, flow-based switches are forwarded based on the network traffic forwarding policy passed from the central controller. This framework laid the foundation for the SDN architecture. For the first time, OpenFlow: Enabling Innovation in Campus Networks published by Stanford University in 2008 shows the working principle and application scenarios of OpenFlow [2]. In 2009, Professor McKeown formally proposed the concept of SDN [4]. For the first time in the OpenIT White Paper published by MIT, SDN describes the changes that the OpenFlow protocol brings to the network. Ten top technologies in 2009 included SDN.

### 2.2 SDN Architecture

Different organizations have released different versions of SDN architecture according to their different needs and were first proposed by the Open Networking Foundation (ONF) [5]. At the same time, this architecture is also the most recognized architecture in various industries such as Figure 1 shows. The structure of SDN from north to south followed by the application plane, control plane, data plane [6]. The data plane is mainly composed of network devices such as switches. It is mainly responsible for data processing, forwarding, and status collection. The paths for interconnecting network data exist between devices to complete data forwarding. The control plane is connected to the data plane through the southbound interface and obtains the device information of the data plane from the controller capable of arranging the data plane resources. Meanwhile, the distributed network of the data plane network device attempts to be abstracted by the controller in this layer Out, to support the SDN controller on the whole network of information unified configuration control. After the controller calls a specific network path through CDPI, the controller provides editable NBIs for

different user needs so as to tailor the application to meet the needs and realize the logical management of the network. The application plane at the top of the SDN architecture mainly includes different types of services and applications, enabling users to rapidly deploy new applications through simple programming and then deliver the data plane forwarding rules through CDPI. The ONF organization defines a southbound interface connecting the two planes according to the location relationship between the data layer and the control plane and defines an open OpenFlow protocol standard [7]. According to the location relationship between the application plane and the control plane, However, due to the huge difference between different business applications, there is still no standard northbound interface protocol standard proposed.

In addition, the NFV architecture proposed by the European Telecommunications Standards Institute (ETSI) [8] released the latest SDN development roadmap by three major companies, Cisco, Juniper Networks and Extreme Networks. And OpenDaylight [9], which is supported by many IT companies. The goal is to promote the deployment and implementation of industry solutions through the development of SDN. Due to the compatibility requirements, OpenDaylight follows the ONN standard architecture of SDN, and OpenDaylight is also integrated into the NFV. The structure of SDN is divided into network application and business process, control platform and physical and virtual network equipment three levels. OpenDaylight comes with a Java virtual machine that can directly implement the SDN control platform. To make it more compatible with third-party modules, the OpenDaylight controller also has plug-in modules that install itself to further enhance the functionality of SDN. The most prominent feature of the OpenDaylight controller is that the southbound interface not only supports the OpenFlow protocol but also supports the configuration protocols and routing protocols such as NETCONF [10] and BGP [11]. Proprietary protocols for each company, such as Cisco's OnePK protocol [12], are also compatible. OpenDaylight adds a layer of abstraction SAL that unifies different underlying protocol standards to ensure scalability and transparency of the underlying protocol.

Table 1 Contrast of SDN, NFV and OpenDaylight

Architecture	ONF SDN	NFV	OpenDaylight
Interface standard	OpenFlow	A variety of interface protocol support scalable	A variety of interface protocol support scalable
Compatible with SDN	—	Can work together	compatible
Application areas	Campus Network, Enterprise Network, Data Center	Operator network	Internet in the future
sponsor	University researchers	Operator	Equipment manufacturers, software vendors
organizer	ONF	ETSI NFV	OpenDaylight
Features	Emphasis CNC separation	Network function virtualization	Software open source and implementation

Table 1 shows the comparison between SDN, NFV and OpenDaylight. Due to the special nature of NFV, it can work synergistically with ONF's SDN architecture to replace the existing hardware with virtualized software, allowing application expansion and device upgrades to be independent and reducing equipment costs. OpenDaylight is an open source architecture that is compatible with both architectures. Although the architectures are not consistent, the overall purpose of each architecture is to separate the data and control of the network for better programmability, allowing users to deploy the network through various means, such as terminals.

### 3. The Key Technologies of SDN Data Plane

In the SDN architecture, the control plane is mainly responsible for the control and strategy formulation. This liberates the data plane in the lower layer so that the task of the data plane simply performs the fast forwarding of the data packets according to the flow table rules delivered by the upper-level control plane. Due to frequent waste of resources and time due to frequent interactions

with the control plane, data plane forwarding is based on stream-level rather than traditional packet-level. Because the task of the data plane is relatively simple, the research on the data plane mainly focuses on the traffic scheduling of the data plane.

At this stage, the researches on traffic scheduling in data plane mostly focus on the application of SDN in data center. Some of the most popular methods are flow-oriented scheduling and flow control to maximize the use of bandwidth. Common stream-oriented scheduling methods are MiceTrap [13], OpenSample [14] and SASCD [15]. The common methods to maximize the network bandwidth are Hedera [16] and Baatdaat [17] Dynamic load balancing algorithm [18] and so on.

### **3.1 Flow-Oriented Traffic Scheduling Method**

In MiceTrap, a stream with low short-time transmission time (mouse flow) is aggregated, and a new load balancing scheme is provided by a software configurable weighted routing algorithm, so as to reduce the number of mice left to achieve more efficient traffic scheduling. OpenSample uses sFlow Technology samples the packet to further enable a more efficient SDN control loop, a low-latency, sampled network traffic platform that can obtain each stream's information under any network load. Compared with the traditional polling method, OpenSample can provide a 100ms control loop, which greatly improves network throughput. SASCD is a data center network architecture designed to analyze network traffic based on SDN data centers The key steps in optimization, using a static flow scheduling genetic algorithm, make the link of network have a better advantage in load balancing. However, once the algorithm undergoes changes in the traffic matrix in the network, it is necessary to recalculate it, which is more difficult to implement in a real environment.

### **3.2 Maximize The Use of Bandwidth Traffic Scheduling Method**

Hedera proposed a new scheme of bandwidth demand, which utilizes the multiple equal-cost paths in the data center combined with the OpenFlow technology to dynamically estimate the large-flow bandwidth. The simulated annealing algorithm is used to calculate the optimal distribution path of the large-flow, dispatching network bandwidth utilization to improve. However, Hedera is only considering unilaterally the placement of large-scale flows, and knowledge has expanded the utilization rate of bandwidth to a certain extent. Without considering the complexity of network traffic in the current network environment, it often results in too many small Baatdaat obtains the utilization of network bandwidth by measuring the utilization of the network and utilizes the remaining bandwidth in the network to alleviate the problem of link congestion so that Baatdaat can dynamically adjust the network bandwidth Traffic, even in the case of network congestion, can still improve the efficiency of stream forwarding. In the dynamic load balancing algorithm of fat-tree network, the greedy algorithm is used to guide the flow to the core of the network and then transmit it to the destination node. However, this algorithm only considers the state of single hop and has insufficient control over the global state of the network.

## **4. The Key Technologies of SDN Control Plane**

The SDN architecture uses a top-down, logical centralized network control [19]. This structure allows the SDN controller to centrally coordinate all the switches in the network. Centralized design brings many benefits to managing the network effectively and responding to dynamic events quickly. There are two ways in which the control plane can be logically concentrated: single and distributed SDN controllers. In a single SDN controller, only one machine SDN controller exists in the network. All switches are centered on it and connected to the same instance. Unfortunately, this approach has two main problems:

- 1) Scalability issues The size of the network and the ability to control large-scale traffic are summarized as scalability. For SDNs, scalability means the controller's ability to handle this information when a large number of switches send path forwarding requests to the controller. To solve this problem, researchers attempt to limit the number of forwarding paths requested to the SDN controller [20,21]. However, this way of controlling the functionality below violates the SDN's design intent.

2) Stability issues. In the single-controller mode above, a problem with this controller means that the data plane's forwarding capability will be completely down and eventually the entire network will be down. To solve this problem, researchers can configure it as an Open Flow hybrid switch [22]. The hybrid switch is a function switch that switches the exchange mode between traditional and SDN. When the controller is disconnected from the switch, the hybrid switch switches the exchange mode to the traditional mode.

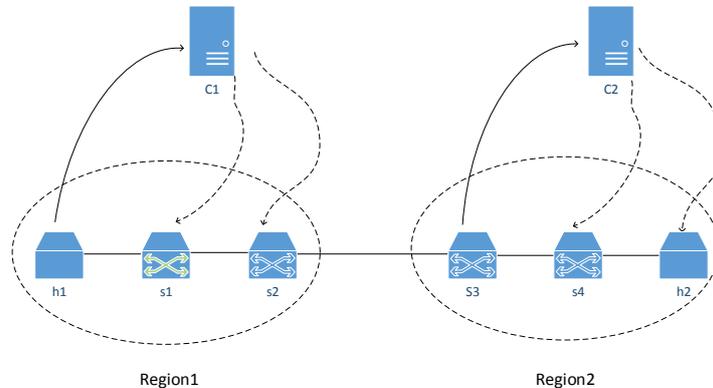


Fig. 2 Multiple controllers work in different domains diagram

The second deployment of multi-controller, distributed controller deployment is more common way to achieve a single controller to solve the two problems exposed by the general approach is to form a load sharing network controller Model, most controllers can act as a substitute for other controllers, and where there is a problem at random, another controller can replace its position. In addition, the distributed SDN controller architecture not only has multiple controllers. The topology shown in Figure 1 consists of two SDN controllers (the solid line is Packet-In route request information and the dotted line is the forwarding rule delivered by Flow Mod). c1 Manages the switches in Domain 1 and manages c2 Now suppose h1 needs to send packets to h2. When the packet reaches s1, s1 requests the forwarding path Packet-In message from c1 to Open Flow. After getting the required forwarding path information, the packet is forwarded to s2. Similarly, when the packet arrives s3. s3 needs to forward the path (ie, re-enter the message through the packet) The final packet arrives at h2. In this case multiple SDN controllers deployed in the network, but they work independently for each network domain. Therefore, two forwarding path requests occur in this case (at s1 and s3), which does not represent a distributed SDN controller architecture.

If c1 and c2 are shared logically and logically so that both c1 and c2 can install the forwarding path directly on all the appropriate switches when the new packet arrives at s1. In this case, the forwarding path request can send the packet directly to the destination as if all the switches were connected to a single SDN controller. This example illustrates the SDN controller, a distributed SDN controller, that logically centers the terms of the physical distribution.

For the distributed deployment of SDN controllers, each of the different design ways will affect the performance of the SDN controller, such as scalability, fault handling, consistency and so on. Table 3 compares the flat and hierarchical distribution of two kinds of controllers in these aspects of the comparison.

Jie Hu et al. [23] studied scalability metrics for SDN controllers in four different controller architectures: a single centralized SDN controller; a distributed SDN controller with a planar architecture (local view, each SDN controller Only local network status), the controller also has local network status from its neighboring controllers but it is abstracted as a single node; a distributed SDN controller (global view) with a planar architecture; a hierarchical distributed SDN controller. From their experimental results, we can see that the single controller is the worst in scalability. The planar distributed SDN controller shows the best scalability in a large number of experiments. The layered model is consistent leading. In a hierarchical model, the number of SDN controllers should be kept to a minimum. The larger number of root controllers may result in a less coordinated network efficiency [24]. This concept is detrimental to the layered model in terms of troubleshooting. The smaller the number of SDN controllers, the smaller the SDN controller can be used to take over the fault

controller. In addition, in a layered model, the SDN controller is divided into two roles (root controller and local controller) with different capabilities. Due to this difference, the local SDN controller may not be able to take over the root controller and vice versa. The root controller may only take over the root controller and the local controller takes over the local controller. As a comparison, the mechanism in the SDN controller failover plane model is more straightforward because all SDN controllers are similar replication state machines and any SDN controller can employ any failed controller in the network. In addition, with a replication state machine, you can build a robust controller failover mechanism that takes into account security and the correctness of the switch and SDN controller during the failover process [25].

In terms of privacy protection, in a hierarchical model, higher-level network organizations should play a role as top-level SDN controllers (for example, root controllers). At the same time, lower-level organizations play a role as subordinate SDN controllers [26]. All local SDN controllers can share abstract local network status to the root SDN controller. The root controller can then form an abstract global network state. With this abstract version of the global network status, the root controller does not know the local details network status. The local controller can not know the local unexpected network information. Therefore, a win-win solution has been resolved.

In a flat model, a controller cluster requires a way to share the abstract network state. One solution is to use network abstraction table [27] This is a form state that maps real network values to abstraction. Each SDN controller must maintain its network abstraction table and share the abstraction of the local network state to other SDN controllers. Then, each of them can be combined with information to form a global state of abstraction, just as in a hierarchical model. The SDN controller must update the network abstraction table periodically on the same state as the local network.

Based on simplicity, the hierarchical model wins because the abstraction can be done simply by design. Therefore, no network abstract table is needed. Therefore, the hierarchical model has advantages. In terms of consistency, only the root controller can grasp the abstract network view of the entire network in the hierarchical model. In planar mode, all SDN controllers maintain the entire network-wide state. Compared with the planar model, the performance of the hierarchical model is slightly less than the consistency.

## 5. SDN Applied Research

A variety of SDN applications are included in the SDN application plane. SDN applications can help users avoid unnecessary and complex programming issues, enabling users to easily deploy new applications on demand. At present, many companies such as Google, Huawei, Cisco and other companies have proposed their own SDN solutions, SDN technology will be applied to their own technology, while SDN in other aspects of the application is also gradually carried out. SDN applications are still in the development stage, mainly used in the optimization of network resources. Including the application in the campus network, the application in the wide area network and in the data center.

### 5.1 SDN in The Campus Network Applications

Campus network is a typical campus network, the traditional campus network structure is generally divided into the core layer, the convergence layer and access layer three levels. The core layer and the convergence layer are connected through static routing protocols such as OSPF. Deploy network timing and intrusion detection facilities on the link trunk. However, there are many problems with the traditional networks because once the above three-layer network structure is deployed, it is very difficult to make any changes. As a result, the campus network is not flexible enough. When the network requirements change, it can not respond in time and network maintenance Function of the maintenance costs are higher.

SDN technology utilizes the technology of numerical control separation to realize the centralized management of logic and increase the programmability of the control plane, laying a good foundation for the simplification of network management and the reduction of maintenance cost. The

standardization of SDN southbound interfaces and the centralized control logic make the SDN control plane more data-oriented traffic scheduling more global optimization, which significantly improves the management performance of campus networks.

The application of SDN in campus network is common for the deployment of dynamic service function chain and the deployment of campus network based on SDN structure.

In the deployment of campus network based on SDN structure, although different school network structures are different, the deployment ideas are basically the same, which are often divided into steps of data forwarding design and control platform deployment. The forwarding strategy in the data forwarding process is implemented on the controller. The data plane switch can choose OpenvSwitch of the virtual switch or select products that support OpenFlow by companies such as Huawei. For the deployment of the control platform, a visual management interface is established according to the current status of campus network deployment and the global view of management, and the global optimization management of the overall network resources is implemented.

### **5.2 SDN in The Wide Area Network Application**

The constant interaction with shared data and private data has made the current demand for agility of the WAN increasingly high. Newer business traffic patterns such as big data and the Internet of Things are changing faster and faster, and their traffic flexibility requirements are also getting higher and higher, which makes the traditional mode of manual configuration in the WAN can not keep up with the market demand, and the flexibility of SDN makes it in the WAN can have a better application.

The most representative of the application of SDN technology is Google company, Google company built SDN-based B4 structure. The use of SDN combined with ECMP hashing technology for global information acquisition and distribution, to ensure the independence of user resources and applications, while ensuring the flow of balance and efficiency. Through the trial in recent years, the structure can almost achieve 100% utilization in resource trial, and at the same time ensure that the use efficiency is maintained at more than 70% for a long time. Google's B4 system is the most successful SDN application.

### **5.3 SDN in The Data Center Application**

With the continuous expansion of network scale, the burden on network links is getting heavier and heavier, and the traffic scheduling between large-scale WANs becomes more and more urgent problems to be solved. At present, the dilemma faced by WAN capacity management in WAN and the low utilization rate of dedicated lines make the demand of users unsatisfied, which is reflected in companies such as Tencent. At the same time, Microsoft also encountered the problem of insufficient link utilization on its backbone network. This makes the WAN in advance planning of traffic and the distinction between the more crucial functions of the diversion.

Tencent Group to Tencent Suzhou cloud, for example, when the Suzhou - Zhengzhou network link changes, Tencent company before the network can be implemented within the control of change, that is to use Tencent cloud routing instead of Suzhou BGP access, so should The flow of network traffic to Suzhou flows into Tencent backbone carrier network, which connects user traffic to Suzhou through the backbone carrier network. Although this avoids the Suzhou user's network connection will not be interrupted, but it will bring a greater burden on other nodes. Tencent's solution to this problem is through refined BGP route management between SDN graphic design provinces: if it is a BGP network interruption from Suzhou to Zhengzhou, the traffic between Zhengzhou and Suzhou is solely switched to other nodes, and the other provinces Of the traffic is still normal access to Suzhou, thus avoiding the other nodes to bring a greater load of the problem.

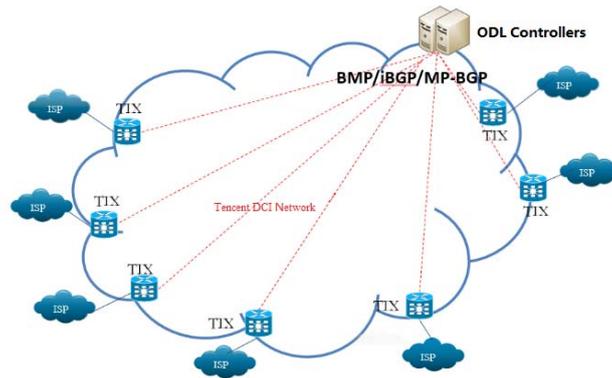


Fig. 3 Tencent DCI network diagram

As shown in Figure 3, all BGP routers in the domain are connected to the SDN controller that controls the global information. The control plane can simultaneously receive the alarm information of the global network and monitor the network situation of the global network. On the other hand, the control plane sends the route and path selection according to the demand to the access routers in the whole country through the SDN controller. Different types of options for different weight and configuration options required, the system pre-set network optimization algorithm. (Such as ISP optimization, ASPATH shortest priority, network resource utilization, network delay optimization, etc.), network scheduling solutions are automatically deployed by manual audit follow-up. Tencent controller deployment through Tencent backbone carrier network connected together to support both High Availability and In-Service Software Upgrade. Flow scheduling between users is all based on ODL controller-based BGP traffic switching at the second level to ensure accurate on-demand, non-perceptual network operation. Tencent currently runs SDN network scheduling technology, the average speed of 30 times / month, the average operation time of each schedule does not exceed 3 hours, but each successful handling of the user network connection problems such as saving more than 70 hours, a substantial increase Tencent's BGP quality cloud network.

## 6. Development Trend

SDN / NFV in 2016 and the SDN / NFV Conference just concluded some time ago, major authoritative organizations, operators and enterprises all focused on SDN deployment issues in specific industries such as carrier networks, enterprise networks and cloud data centers . Making SDN is no longer limited to just one concept, but the market in the urgent need for the implementation of SDN in the industry, which also indicates the arrival of SDN golden age. SDN future trends can be summarized as the following aspects:

### 6.1 SDN and NFV Combination of Applications

NFV As SDN's "twin brother," the work focuses on the upper layers of the web. It covers network functions such as routers, firewalls, and load balancing. However, NFV does not depend on SDNs and can promote each other between NFVs and SDNs. The combination of SDN and NFV enables the best delivery of network services: software for network-specific devices and virtualization of network functions through NFV. The future of SDN and NFV solutions will be closely integrated and develop greater value.

### 6.2 SD-WAN Technology

By utilizing virtualization technology, application-level policies and Overlay networks and other devices, SD-WAN enables the utilization of multiple public or private links so that common links can achieve leased line QoS, reducing costs and increasing network bandwidth. Use the existing network conditions and configuration automatically select the best strategy for load balancing. The evolution of SD-WAN technology over the past two years has focused mainly on improving managed services, and future SD-WAN technologies will include the ability to co-host services across service providers, while at the same time better performing with other web applications and open APIs Integrated development to better support virtual client devices.

### 6.3 SDN 5G May Get the Application

The 5G standard has not yet been set, but 5G has put forward some due diligence in the industry: reliability, flexibility and scalability. Compared with the current 4G technology requirements, 5G technology can provide more users with services, greater data traffic and bandwidth, and shorter end-to-end delay. However, none of the characteristics of SDN meets the requirements of 5G proposed by the network, and thus it can be concluded that SDN technology may also play an important role in 5G technology in the future.

### 6.4 SDN in Network Security Applications

The complexity of the current network situation has led to various network problems. SDN technology verifies the source address by using a programmable controller, which can effectively trace the source address-based cheating network attacks. The design and implementation of firewall based on SDN controller is also the hot topic in current research. However, the complexity of the network also determines the diversity of network risks. At present, the SDN technology is still in the development stage. The research center is also mainly focused on the reconstruction of the network architecture and the optimization and utilization of network resources. In the future, there will surely be further development of.

## 7. Conclusion

SDN is one of the more popular and cutting-edge technologies of the moment, and its unique architecture makes it an advantage that traditional networks can not. However, if SDN technology wants to get better development and application in the future, further research and optimization must be made on cost control and security. This article takes the structure of SDN as an entry point, introduces the research status of each SDN plane and the application of SDN in various fields respectively. Finally, it forecasts the future development trend of SDN and the possible application direction.

## References

- [1]. Stanford University. Clean slate program. 2006. <http://cleanslate.stanford.edu/>
- [2]. McKeown N, Anderson T, Balakrishnan H, et al. OpenFlow: enabling innovation in campus networks[J]. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 69-74.
- [3]. MIT Technology Review. 10 breakthrough technologies, TR10: Software-defined networking. 2009. <http://www2.technologyreview.com/article/412194/tr10-software-defined-networking/>
- [4]. McKeown N. Software-Defined networking. In: Proc. of the INFOCOM Key Note. 2009. <http://infocom2009.ieee-infocom.org/technicalProgram.htm>
- [5]. Open networking foundation. 2014. <https://www.opennetworking.org/>
- [6]. Open Networking Foundation. Software-Defined networking: The new norm for networks. ONF White Paper, 2012
- [7]. McKeown N, Turner J. OpenFlow: Enabling innovation in campus networks. ACM SIGCOMM Computer Communication Review, 2008, 38(2): 69, 74. [doi: 10.1145/1355734.1355746]
- [8]. ETSI. Network functions virtualisation. NFV White Paper, 2012.
- [9]. OpenDayLight. 2014. <http://www.opendaylight.org/>
- [10]. Enns R, Bjorklund M, Schoenwaelder J, Bierman A. Network configuration protocol (NETCONF). IETF RFC 6241, 2011.
- [11]. Rekhter Y, Li T, Hares S. A border gateway protocol 4 (BGP-4). IETF RFC 4271, 2006.
- [12]. Cisco. Cisco's one platform kit 2014. <http://www.cisco.com/en/US/prod/iOSSwrel/onepk.html>

- [13]. Trestian R, Muntean G M, Katrinis K. MiceTrap: Scalable Traffic Engineering of Datacenter Mice Flows using OpenFlow[C]// Integrated Network Management, Ifip/ieee International Symposium on. IEEE, 2013:904 - 907.
- [14]. Suh J, Kwon T T, Dixon C, et al. OpenSample: A Low-Latency, Sampling-Based Measurement Platform for Commodity SDN[C]//Distributed Computing Systems (ICDCS), 2014 IEEE 34th International Conference on. IEEE, 2014: 228-237.
- [15]. Yang Bo Research on Data Center Network Traffic Optimization Based on SDN [D]. Chang Sha: National University of Defense Technology 2014
- [16]. Al-Fares M, Radhakrishnan S, Raghavan B, et al. Hedera: dynamic flow scheduling for data center networks[C]// Proceedings of the 7th USENIX conference on Networked systems design and implementation. USENIX Association, 2010:19-19.
- [17]. Tso F P, Pezaros D P. Baatdaat: Measurement-based flow scheduling for cloud data centers[C]// 2013 IEEE Symposium on Computers and Communications (ISCC). IEEE Computer Society, 2013:000765-000770.
- [18]. Li Y, Pan D. OpenFlow based load balancing for Fat-Tree networks with multipath support[C]//Proc. 12th IEEE International Conference on Communications (ICC,13), Budapest, Hungary. 2013: 1-5.
- [19]. ONF. (2012, July 27 2015). Software-Defined Networking:The New Norm for Networks. [White Paper].
- [20]. A. R. Curtis,P.Sharma."DevoFlow: Scaling flowmanagement for high-performance networks," in ACM SIGCOMM Computer Communication Review, 2011, pp.254-265.
- [21]. M. Yu, J. Rexford, M. J. Freedman, and J. Wang, "Scalableflow-based networking with DIFANE," ACM SIGCOMMComputer Communication Review, vol. 40, p. 351, 2010.
- [22]. ONF. (2011, July 28 2015). OpenFlow Switch Specification Version 1.1.
- [23]. J. Hu, C. Lin, X. Li, and J. Huang, "Scalability of control planes for Software defined networks: Modeling and evaluation," in Quality of Service (IWQoS), 2014 IEEE 22nd International Symposium of, 2014, pp. 147-152.
- [24]. D. Marconett and S. Yoo, "FlowBroker: A Software Defined Network Controller Architecture for Multi Domain Brokering and Reputation," Journal of Network and Systems Management, vol. 23, pp. 328-359, 2014.
- [25]. N. Katta, H. Zhang, M. Freedman, and J. Rexford, "Ravana: Controller Fault-Tolerance in Software-Defined Networking," presented at the SOSR 2015, Santa Clara,CA, USA., 2015.
- [26]. B. Lee, S. H. Park, J. Shin, and S. Yang, "IRIS: The Openflow based Recursive SDN controller," in Advanced Communication Technology (ICACT), 2014 16th International Conference on, 2014, pp. 1227-1231.
- [27]. P. Lin, J. Bi, and Y. Wang, "East-West Bridge for SDN Network Peering," Frontiers in Internet Technologies, pp. 170-181, 2013.