An Idea for Improvising the Efficiency of SDN Based Business Design with SOA

M. Priyanka and K. John Singh
School of Information Technology and Engineering
VIT University, Vellore, Tamil Nadu, India.

Abstract: In today’s world the business environment is experiencing the dramatic change in the field of computer networking. Global competitions have forced traditional and non-traditional organization to be more aggressively influence the technology for competitive advantage. But many organization have struggled to manage the changes in order accelerate time to market with new technology. Equipment manufacturers frequently need to adapt new technologies and change their product portfolio. This level of change will be a significant factor in the success of organizations. The trends in software defined networking and increased dependency on software solutions derive the changes in market of infrastructure networking and provide the centralized service and control. The aim of the proposed work is to resolve many problems of traditional networks such as manufacturer dependency, non open protocols, and closed network architecture, time to market and open community for network architecture.

Index terms: controller, data, SOA, SDN.

I. INTRODUCTION

In traditional networking application, the network requirements are described indirectly and had several processing steps. The way of expressing the user requirements are limited and non dynamic in nature. Manufacturers try to infer the user requirements on their own, which may lead to increased infrastructure cost, cost of development over new architectural design and time to market on new protocol designs. Software Defined Networking provides the monitored relationship and offers the ability for user to specify their needs. “The main aim of SDN is to provide open interfaces enabling development of software that can control the connectivity provided by a set of network resources and the flow of network traffic though them”. Fig.1 Architecture diagram of SDN explain the data, control and application plane. Data Plane is comprised of network elements, whose SDN Data paths expose their capabilities through Control-Data-Plane Interface (CDPI) Agent. SDN Application exists in the application plane and communicates their requirements via Northbound Interface Drivers. In center the SDN controller translates these requirements and exerts low level control over the SDN data paths, while providing relevant information up to the SDN applications. The management and admin plane is responsible for setting up the network elements, assigning the SDN data paths their SDN controller, and configuring policies defining the scope of control given to the SDN controller or SDN application. This SDN network architecture can coexist with a non-SDN network, especially for the purpose of migration to a fully enabled SDN Network.

Open flow is the standard communication interface defined between the controls and forwarding layers of SDN architecture. Open flow allows direct access to manipulation of the forwarding plane of network devices. Benefits that enterprise and carriers can achieve through open flow based SDN architecture include

- Centralized control of multi vendor environments
- Reduced complexity through automation
- High rate of innovation
- Increased network reliability and security.
- Better user experience
The design of a future internetworking architecture can benefit from software architecture paradigms to make a more flexible and easy to maintain network architecture rather than having an inflexible architecture. Some of the argument show how the SOA is suitable for internetworking architecture are service contract, loose coupling, abstraction, reusability, autonomy, statelessness, discoverability, compositability.

The internetworking architecture should be flexible, network should be able to adapt to specific customer or application needs and changing environmental conditions. And network should be able to evolve, meaning that functionality is added, changed or even removed. It is then achieved by composing several smaller services into more complex and specialized services. SOA aim to provide a dynamic composition of service.

By implementing these two concepts in to one model, we can be able to avoid the difficult faced by user. User has to rely on the vendor to modify the protocol which is being embedded internally. Instead of being handled by API we can even classify our processing step and provide it as a service so that user can develop protocols by the open community and restricts the development to the network equipment manufacturing companies. In this...
section we described the brief about the recent trend of service and how the concept can be utilized further. In section II, we described the existing work and related survey for the efficiency of SDN based business design with SOA. We described the materials required and the details of the proposed approach in section III. In section IV, we concluded the paper.

II. RELATED WORK

Paul Mueller et al. [1] gives an idea of design foundation for future network architectures there are still several open issues remain for applying the SOA paradigm in network architectures. This paper also argues that short term and long term flexibility is the key for designing new internetworking architectures. To achieve such a required flexibility it is necessary to overcome the barriers between “distributed-systems” and “networked-system” research based on the ideas of software defined networking. Taking this into account, our approach for a new and flexible internetworking architecture is based on a new software engineering methodology namely the service-oriented architecture principles. Such an approach of fine-grained functionality can enable more flexible networks, so that the network can adapt to new application’s needs and new network capabilities and/or constraints. This also allows the introduction, alteration, and removal of functionalities. Moreover, as any certain functionality is not related to a fixed layer but can be integrated and offered as a new user oriented, application oriented, or network Oriented service when needed (potentially by a third party), this approach may also offer new business models.

ONF White Paper [2] explains the future of networking will rely more and more on software, which will accelerate the pace of innovation for networks as it has in the computing and storage domains. SDN promises to transform today’s static networks into flexible, programmable platforms with the intelligence to allocate resources dynamically, the scale to support enormous data centers and the virtualization needed to support dynamic, highly automated, and secure cloud environments. An SDN approach fosters network virtualization, enabling IT staff to manage their servers, applications, storage, and networks with a common approach and tool set. Whether in a carrier environment or enterprise data center and campus, SDN adoption can improve network manageability, scalability, and agility. SDN is on the way to becoming the new norm for networks.

Adrian Lara et al. [3] have proposed the idea for two important challenges faced by network managers. First, the deployment of middle boxes in choke points of the network raises robustness, correctness and efficiency concerns. Second, dynamic updates of the network topology require tedious tasks. To solve the task using Software Defined Networking and Open Flow to simplify network management addressing these two challenges. SDN consists of decoupling the control and data planes of a network. Open Flow standardizes the way that the controller communicates with the network devices in SDN architecture.

III. PROTOCOL AS SERVICE-PAAS

Our proposed method aims to improve the efficiency of SDN based business design with service oriented architecture. Let us consider an example on the implementation of the existing OSPF – “Protocol as a Service “in the Software Defined Network architecture. Open Shortest Path First (OSPF) is a link-state routing protocol for Internet Protocol (IP) networks. It uses a link state routing algorithm and falls into the group of interior routing protocols, operating within a single autonomous system (AS). OSPF is perhaps the most widely used interior gateway protocol (IGP) in large enterprise networks. The Principle of the OSPF is based on 5 Packets defined in the RFC 5340 OSPF v3 Packet Format standard.

| Octet | Bit | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 |
| 0     | 0   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 4     | 32  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 8     | 64  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |
| 12    | 96  |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |   |

**Fig. 3** OSPF packet Format
The 5 Packets defined are: The Hello Packet, Database Description, Link State Request, Link State Update and Link State Acknowledgement. These Packets and the Protocol are defined and coded inside a Router. And often these OSPF is just made available as a Configuration. The User can only use these Configuration but cannot change the Protocol itself.

For example, let's consider a vendor Router, in order to enable the OSPF in a router we need to configure the router using:

1. `enable`
2. `configure terminal`
3. `router ospf process-id`
4. `network ip-address wildcard-mask area area-id`
5. `end`

If the User or a developer wanted to develop a new OSPF Algorithm it is highly difficult and the User has to rely on the Vendor to modify the Protocol and embed it onto the router’s internal Chip. This avoids the development of open source protocols by the open community and restricts the development to the Network equipment manufacturing companies.

By implementing the “Protocol as a Service” in Software Defined Network Framework we can easily integrate the Protocols and implement it in the network. Consider the PaaS + SDN in the above case of OSPF. Since the Protocol is just a API or Program Application, we can easily modify the codes of the Implemented API and create a new OSPF protocol which is as per our requirements. This will also lead to the development of more efficient and vast protocols by the Open community just like GNU Linux development.
IV. CONCLUSION

The future of networking will rely more and more on software, which will accelerate the speed of improvement for networks as it has in the computing and storage domains. SDN promises to transform today’s static networks into flexible and dynamic, programmable platforms with the intelligence to allocate resources dynamically, the scale to support enormous data centers and the virtualization needed to support dynamic, highly automated, and secure network environments. With its many advantages and features, SDN is on the way to becoming the new advance trend networks.

REFERENCES

