

The Method of Computing Organization in High Loaded SDN Controller System

Skulysh Mariia

The National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute", UKRAINE, Kyiv, Peremogi Avenue, 37,

Email: mskulysh@gmail.com

Abstract: Using the cloud for software defined networks opens new possibilities for the organization of computing processes of network controllers. The article considers the possibility of using an "endless train" method for virtualization of individual functional opportunities of SDN OpenDayLight controller. This method allows organizing controller computer processes caused by the need to serve the dynamically changing stream of user applications, such as initialization of the flow or the organization of virtual networks.

Keywords: Software-defined network, SDN controller, OpenDayLight controller, Quality of Service, OpenFlow.

I. INTRODUCTION

In recent years a rapid increase in mobile traffic, and change of its structure have been seen. Modern mobile network should be able to support a large number of subscribers and process large amounts of data. They need to create high-performance clusters and well scalable virtualized environment. All these factors lead to an increase in requirements for network environment and it is often the limiting factor in the development of computing infrastructure.

The main problem of today's mobile networks is that they are too static, therefore they do not meet the current dynamics of the allocation process, unlike servers that are required by this virtualization technology. [1] Nowadays applications are distributed between multiple virtual machines that exchange data very intensively (leading to an increase of east-west traffic, which begins to dominate over traditional client-server north-south traffic). Virtual machines often migrate to optimize the load of servers. [2] It modifies traffic binding points. Traditional addressing scheme, logical separation of networks and methods of appointment rules processing traffic in these dynamic environments become ineffective.

Similar difficulties arise with the reconfiguration mechanisms for Quality of Service (QoS) while adding a new multiservice network application, such as video. Procedures of security settings modification in large networks take too much time that prevents quickly respond to emerging threats. [3]

Introduction of technologies of software defined networks and network functions virtualization may be just the factor that would solve existing problems and radically change the approach to the organization and management of the network.

Quality functioning indexes of software defined networks depend on the performance of software modules

that implement network functions. It is assumed that the use of cloud technology will provide SDN with necessary computing resources. Today it is possible to evaluate the degree of utilization of computing resources for the operation of the software controller only theoretically. The method of endless train, proposed in this article, will organize computer processes of software defined networks in such a way that fully use the capabilities of modern computing environment.

II. ARCHITECTURE AND BASIC FEATURES SDN CONTROLLER

Software Defined Network (SDN) is the data network, in which the level of network management is separated from the data transmission device and implemented in software, it is a form of virtualization of computing resources.

Data is transmitted according to the routing tables stored on hardware systems as according to the existing approaches. But these tables are centrally managed by remote system, therefore it is necessary to modify the table on each switch individually. Ideally, all network components must be controlled and adjusted during one operation. Collaboration of software network components can be caused by standards-based OpenFlow.

Key principles of SDN are separation of transmission and management data processes; centralized network management by using standardized software and virtualization of physical network resources. OpenFlow protocol, which implements an independent interface between the logic controller and transport network, is one of the implementations of software defined network. [4]

SDN main idea is to separate the traffic management and transfer functions (including control of the traffic and of the devices that carry its transfer). In traditional switches and routers, these processes are inseparable from each other and implemented in a "box": special chip network equipment providing forwarding packets from one port to another, and overlying software determines the rules of the transfer, performs the necessary analysis package and changes service information in it, and so on. To determine the route of transmission or prevent looping traffic devices certainly exchange data, with the help of different protocols such as Open Shortest Path First (OSPF), Border Gateway protocol (BGP) and Spanning Tree, but each device operates quite autonomously. [5]

Fig. 1 shows the concept of SDN, according to which all logic control is made in the so-called controllers that are able to monitor the entire network.

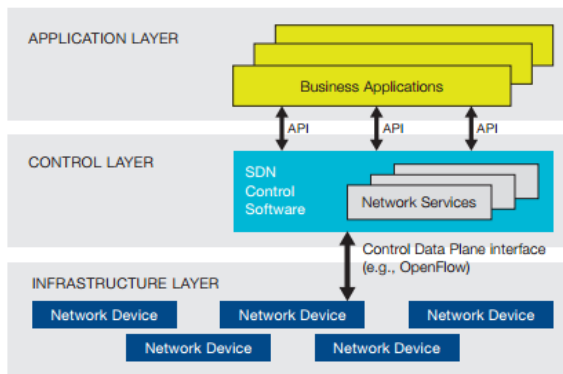


Fig. 1 The concept of software defined network

The main driving force of the concept of software defined network and basic element of its architecture is OpenFlow. OpenFlow is a protocol, which purpose is to manage network devices with SDN-controller. It provides an opportunity for direct programming of network equipment (such as switches and routers), both physical and virtual, making the network more dynamic and controlled.

The main characteristic of OpenFlow is the use of streams for identifying network traffic. These streams are based on pre-defined rules that can be statically or dynamically programmed using SDN-controller. [6]

Consider in more detail the components of SDN controller architecture in order to highlight features that require variable number of resources for implementation.

SDN controller architecture consists of three levels:

1. Southbound plugins and protocols forming the network device layer.
2. Service adaptation and network functions forming the coordination and control layer.
3. Northbound APIs and applications forming the application layer.

Fig. 2 shows an enhanced architecture of OpenDayLight controller, further, describe all the important components of the architecture.

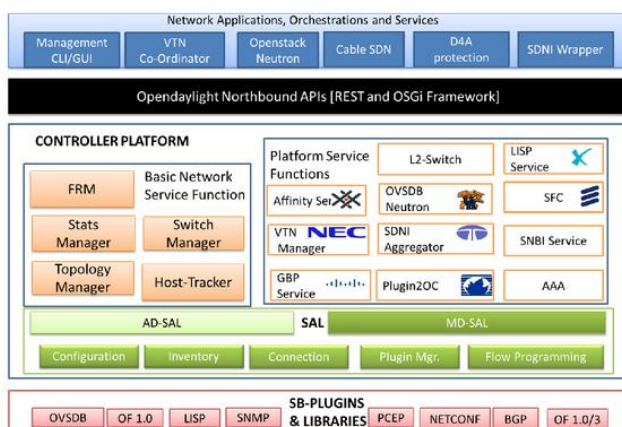


Fig. 2 Enhanced architecture of OpenDayLight controller

Southbound Protocols. The southbound interface is capable of supporting multiple protocols, e.g. OpenFlow 1.0, OpenFlow 1.3, BGP-LS, LISP, SNMP, etc. These modules

are dynamically linked to a service abstraction layer, which determines how to fulfill the service requested irrespective of the underlying protocol used between the controller and the network devices.

Service Abstraction Layer. In OpenDaylight, the service abstraction layer (SAL) is the key design that enables the abstraction of services between the services' consumers and producers. SAL acts like a large registry of services advertised by various modules and binds them to the applications that require them. Modules providing services, or producers, can register their APIs with the registry.

Service Functions. The controller has various basic network functions included as part of the shipped base. This includes services for topology discovery and dissemination, a forwarding manager for managing basic forwarding rules, and a switch manager for identifying networking elements in the underlying physical topology. The basic service functions are:

1. Topology – a service for learning the network layout by subscribing to events of node addition or removal. Application requiring network view can use this service.
2. Statistics – service for managing stats counters across nodes, flows and queues (ports).
3. Switch Manager – along with the Topology Manager, responsible for storing the nodes discovered on the physical layer.
4. Forwarding – in conjunction with Switch Manager, can provide service for registering and maintaining network flow state. Applications using this can be agnostic to network device specific.

III. THE METHOD OF "ENDLESS TRAIN" FOR THE ORGANIZATION OF THE COMPUTATIONAL PROCESS SDN

Today, there are several approaches to organization of computational process in SDN controller. Their main disadvantage is that they do not consider the possibility of unlimited resources which provide cloud technologies. All existing approaches based on the fact that the amount of system resources is limited and depends on the technical capabilities of the server that hosted virtual machines.

The result of the application of these approaches is reducing of the quality of service at certain time periods that are related to the work of the monitoring system is to collect and analyze data about the network quality of service. If the QoS parameters for certain services are below the threshold, but other services are provided with a sufficiently high quality, the system cannot react to lower quality in a certain period of time, counted as only an average performance.

The solution to this problem is the use of heterogeneous cloud environments, namely certain decisions for platform as infrastructure (PaaS). Platform-as-Infrastructure is an isolated cluster, consisting of a group of servers and services that work together as a complete system, allowing easy deployment, test, maintain and scale the system. PaaS lets you create and maintain an unlimited number of virtual machines whose job is to serve the computing functions of

the controller. The use of an unlimited number of resources for virtual machines will avoid periods of reduced quality of service.

The method proposed in this article is called "effect of endless train". Its main idea is that after a certain virtual machine for service received a specified number of tasks, a new virtual machine will be created, which receives all of the following tasks (fig. 3).

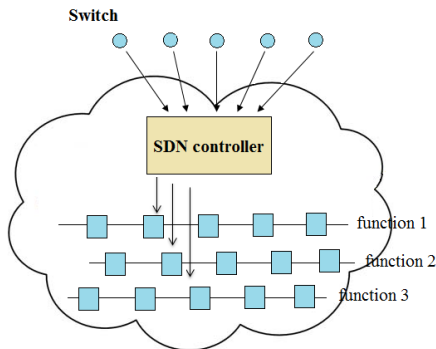


Fig. 3 The method of "endless train"

The proposed method is based on the method of dynamic migration of virtual machines, developed by Jelastec [8], which provides server load balancing cloud storage by creating a platform for the automatic control of virtual machine containers. Also considered the method of choice container for migration of virtual machines described in [9], which allows you to analyze and predict the load on the network based on an assessment of resources used.

The use of cloud storage provides that all data is stored on a large number of distributed servers in the network. Two basic tasks must be performed:

- Interactive division of client tasks between virtual machines that are located in one or more clusters. On the one hand it is the task of load balancing, on the other - the task of ensuring the reliability of service;
- Monitoring and management of a cluster of virtual machines. Cluster resources should always be enough for all virtual machines that run on all servers in the cluster at the same time.

The method of "endless train" implies that all requests for service are sent to the current virtual machine as long as it is filled. The number of applications that can be processed by one virtual machine depends on the amount of resources that is allocated to it at creation. After completing this machine, a new one, which will accept all subsequent applications, will be created. Using this method is possible if the amount of system resources is relatively unlimited. The use of heterogeneous cloud environment provides this opportunity. [10]

Old virtual machines continue to serve their information flows as long as their number does not run out. Then virtual machines are fold, or wait for re-commissioning as "empty wagon." The maximum number of requests that can be served in the "wagon", depends on the configuration of the cloud platform, or can be obtained by experience. The number of requests depends on the resource volume that controller uses to perform computing tasks. Flexibility of migration

processes - technical processes of virtual machines maintenance, is also affects maximum number of requests. Illustration of described method is shown in fig. 4.

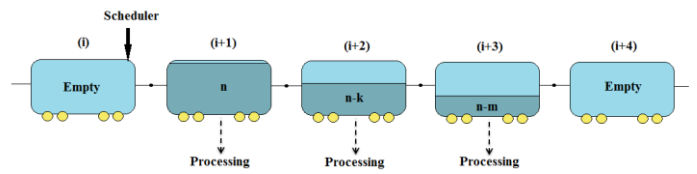


Fig.4. The effect of "endless train"

n - maximum number of tasks for a single VM;

k - number of tasks managed to be processed in the

$(i + 2)$ virtual machine since its start of filling to the current moment – the moment of completion for applications in a virtual machine $(i + 1)$.

m - number of tasks that managed to be processed in the $(i + 3)$ virtual machine from the start of filling to the current moment.

The method of "infinite desire" is a link between level of abstraction and level of computational procedures. The functions of the controller, which implementation requires interaction with a database, you may be considered separate tasks, independent of each other. These functions can be implemented for a given method. These individual tasks can be made in the cloud environment to improve the efficiency of the controller by increasing the amount of computing resources.

The main components of the described system is a set of basic controller functions and a large number of applications that use these functions. Application of the proposed method is appropriate when the system consists of the flow of applications need to be served.

Consider the possibility of using "endless train" method for each of the service functions of the controller.

Topology. Topologies used to store information about network scheme necessary for some applications. The scheme is based on network activity information subscribers exchanged between network nodes. Data from units recorded in a database that is constantly updated. Thus, the function of topology interacts with the database, so the use of endless train method is not considered for it.

Statistics. All applications that work with the function of statistics require computing calculations. The work of statistics function includes processing large amounts of information (data on the network status and activity), execution of computing operations to the received data (calculation of probability and average delay, queue length, etc.) and forecasting network activity based on data obtained two previous stages. Thus, this feature is a stream of requests to perform certain operations that do not require interaction with the database. Therefore, it is appropriate to use the method of "endless train" to increase the speed of processing applications.

Switch Manager. This function is responsible for building a logical network architecture based on statistics function. Physical network topology is used to identify network elements. The method of "endless train" is not applicable for this function.

Forwarding. Forwarding involves searching of the shortest path for the transmission of data flow, analysis of information about reserves and other flow control transactions. The main objectives of this function is the initiation of flow and finding the best way for it. These tasks involve the simultaneous maintenance of a large number of applications and require different execution of computing operations. Therefore, to optimize the flow manager can apply the method of "endless train".

Some vendor components of the controller can also be made in a cloud environment. This applies to the following platform-oriented services as coordinator of Virtual Tenant Network (VTN) and routing protocol BGP. BGP routing protocol is used by SDN controller as protocol management platform for interaction between physical routers, switches and network services. This module searches for the shortest path for the flow of applications. Thus, each application must refer to it. Therefore, these two components are advisable to implement the method of "endless train" and move some of their functionality modules in the cloud environment.

IV. DESCRIPTION OF THE EXPERIMENT

To assess the effectiveness of the proposed method the simulation in GPSS environment was conducted. The delay in servicing requests authentication flow, as well as requests to determine the optimal transmission of information flow was estimated. Three methods of controllers work organization were simulated. Namely:

- Centralized planning of controllers load. This method was modeled as a queuing system with limited amount of service devices;
- Cluster controller organization was simplified simulated as the system of queuing with limited amount of service devices, for which it was established label control delay. If specified conditions are not fulfilled, the load between servicing devices is redistributed;
- Method of "endless train" was simulated as the system of queuing with unlimited (very high) amount of service devices, each function processed by a separate group of service devices.

The result was obtained dependence (fig. 5).

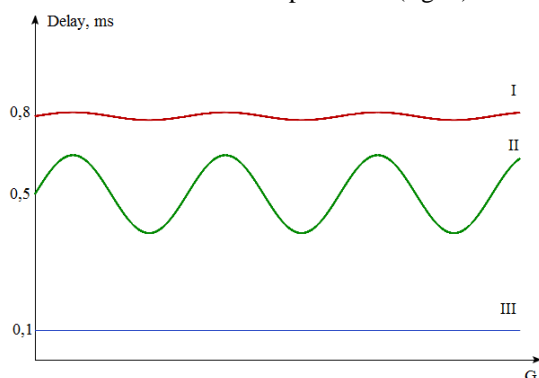


Fig. 5 Experimental time of delay for three different methods

The graph shows that for the method of infinite train delay is the smallest. At the same time limitation of this

model is the amount of resources used for maintenance. This article has not been investigated excessive amount of resources used for initialization and operation of the excess number of virtual machines.

V. CONCLUSION

To expand computing power of SDN controller it is advisable to use cloud technologies. Removal of all functions of the controller or part of them in the cloud environment will allow to operate conditionally endless resource flows to service requests. It is necessary to consider features of virtual service of entities not only in software but also on the physical level.

This article analyzes the possibility of using cloud technology to increase the amount of computing resources of software defined networks. The method of computing processes service in cloud environments called "endless train," which allows to split the flow of applications for individual units that served in separate virtual machines was proposed. This method allows to avoid the additional delays due to overload computer resources and to improve the network quality of service. The effectiveness of the method was evaluated by means of simulation modeling of the network.

REFERENCES

- [1] M. Harris, "Traditional Network Infrastructure Model and Problems Associated with it", *Journal of Pluribus Networks*, vol. 14, no. 2, 2012.
- [2] J. Jinkyu, K. Sung-Hun, K. Hwanju, "Analysis of virtual machine live-migration as a method for power-capping", *The Journal of Supercomputing*, vol. 66, no. 3, pp. 1629-1655, 2013.
- [3] R. L. Smelianskiy Software-defined networks. *Open systems DBMS*, no. 9, pp. 23-26, 2012.
- [4] N. McKeown, "Openflow: Enabling innovation in campus networks", *SIGCOMM Computer Communication Review*, vol. 38, no. 2, pp. 69-74, 2008.
- [5] M. Chiosi, "Network Functions Virtualization: Network Operator Perspectives on Industry Progress", *Proceedings of SDN and OpenFlow World Congress*, pp. 1-16, 2013.
- [6] A. Lara, "Network Innovation Using OpenFlow: A Survey", *IEEE Communications Surv. Tutorial*, vol. 16, pp. 1-20, 2013.
- [7] R. Sridhar, "SDN Series Part Six: OpenDaylight, the Most Documented Controller", *SDN Controller Vendors*, 2015.
- [8] K. Ye., X. Jiang, D. Huang Live, "Migration of Multiple Virtual Machines with Resource Reservation in Cloud computing", *IEEE International Symposium*, pp. 267-274, 2013.
- [9] C. Pahl, H. Xiong, "Migration to PaaS clouds – Migration process and architectural concerns", *IEEE International Symposium*, pp.86-91, 2013.