International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Y.3300

(06/2014)

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

Future networks

Framework of software-defined networking

Recommendation ITU-T Y.3300

ITU-T

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS

GLOBAL INFORMATION INFRASTRUCTURE	
General	Y.100–Y.199
Services, applications and middleware	Y.200–Y.299
Network aspects	Y.300–Y.399
Interfaces and protocols	Y.400–Y.499
Numbering, addressing and naming	Y.500–Y.599
Operation, administration and maintenance	Y.600–Y.699
Security	Y.700–Y.799
Performances	Y.800–Y.899
INTERNET PROTOCOL ASPECTS	
General	Y.1000–Y.1099
Services and applications	Y.1100–Y.1199
Architecture, access, network capabilities and resource management	Y.1200–Y.1299
Transport	Y.1300–Y.1399
Interworking	Y.1400–Y.1499
Quality of service and network performance	Y.1500–Y.1599
Signalling	Y.1600–Y.1699
Operation, administration and maintenance	Y.1700–Y.1799
Charging	Y.1800–Y.1899
IPTV over NGN	Y.1900–Y.1999
NEXT GENERATION NETWORKS	
Frameworks and functional architecture models	Y.2000–Y.2099
Quality of Service and performance	Y.2100–Y.2199
Service aspects: Service capabilities and service architecture	Y.2200–Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250–Y.2299
Enhancements to NGN	Y.2300–Y.2399
Network management	Y.2400–Y.2499
Network control architectures and protocols	Y.2500–Y.2599
Packet-based Networks	Y.2600–Y.2699
Security	Y.2700–Y.2799
Generalized mobility	Y.2800–Y.2899
Carrier grade open environment	Y.2900–Y.2999
FUTURE NETWORKS	**Y.3000–Y.3499**
CLOUD COMPUTING	Y.3500–Y.3999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T Y.3300

Framework of software-defined networking

Summary

Recommendation ITU-T Y.3300 describes the framework of software-defined networking (SDN) to specify fundamentals of SDN. The definitions, objectives, high-level capabilities, requirements and high-level architecture of SDN are addressed in this Recommendation. Appendix I describes areas for further consideration in SDN standardization.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T Y.3300	2014-06-06	13	11.1002/1000/12168

Keywords

SDN, software-defined networking.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T Y.3300

Framework of software-defined networking

1 Scope

This Recommendation describes the framework of software-defined networking (SDN) by providing:

– definitions,

– objectives,

– high-level capabilities,

– requirements, and

– high-level architecture

of the fundamentals of SDN.

In addition, Appendix I describes areas for further consideration in SDN standardization.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T Y.3001] Recommendation ITU-T Y.3001 (2011), *Future Networks: Objectives and Design Goals.*

[ITU-T Y.3011] Recommendation ITU-T Y.3011 (2012), *Framework of network virtualization for Future Networks.*

[ITU-T M.3400] Recommendation ITU-T M.3400 (2000), *TMN management functions.*

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following term defined elsewhere:

3.1.1 network virtualization [ITU-T Y.3011]: A technology that enables the creation of logically isolated network partitions over shared physical networks so that heterogeneous collection of multiple virtual networks can simultaneously coexist over the shared networks. This includes the aggregation of multiple resources in a provider and appearing as a single resource.

3.2 Terms defined in this Recommendation

This Recommendation defines the following term:

3.2.1 software-defined networking: A set of techniques that enables to directly program, orchestrate, control and manage network resources, which facilitates the design, delivery and operation of network services in a dynamic and scalable manner.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

AAA	Authentication, Authorization, and Accounting
FCAPS	Fault, Configuration, Accounting, Performance, and Security
OSI	Open System Interconnection
SDN	Software-Defined Networking

5 Conventions

This Recommendation uses the following conventions:

The term "is required to" indicates a requirement which must be strictly followed, and from which no deviation is permitted, if conformance to this Recommendation is to be claimed.

The term "is recommended" indicates a requirement which is recommended, but which is not absolutely required. Thus, this requirement need not be present to claim conformance.

6 Introduction

Due to the great success of communication technologies, diverse applications are realized on networks, and the requirements that they bring are diverging. To support these diverging requirements, it is necessary to make networks even more controllable and manageable. The necessity to treat different traffic in different manners, and to make networks more application- and service-oriented is also increasing.

Technologies that enable more direct and lower-level (e.g., flow-level) control methods of data packet forwarding are emerging. These technologies can simplify the reaction of network resources (e.g., switches or routers) to operations, and significantly increase the controllability of networks for operators. With modeling and programing of network resources, networks can be controlled in an automated manner, which enables more agile operations of the networks.

This change of control method can also provide a chance to redesign control functionalities by introducing logically centralized and programmable control of network resources through standardized interfaces and protocols.

This approach allows for:

– logically centralized network control, which decreases the number of points to control and manage;
– supporting network virtualization as one important feature of the network architecture;
– defining, controlling and managing network resources using software; thus, allowing network services to be provided in a deterministic manner in accordance with the requested behaviour; and
– network customization, which is necessary for efficient and effective network deployment and operations.

In order to realize the aforementioned features, this Recommendation provides the framework of software-defined networking (SDN) by specifying the fundamentals of SDN with its definitions, objectives, high-level capabilities, requirements and high-level architecture.

While there have been various efforts to develop SDN-related technologies and standards (see [ITU-T Y.3001], [ITU-T Y.3011], [b-ITU-T Y.2622], [b-ETSI NFV], [b-IETF I2RS] [b-IETF RFC 3746], [b-ONF] and [b-OpenDayLight]) with different approaches and different focuses, they all share the same objective of providing the programmability of network resources as described above, which is a core technology for the networks of the future.

7 Overview

SDN is a set of techniques that enables users to directly program, orchestrate, control and manage network resources, which facilitates the design, delivery and operation of network services in a dynamic and scalable manner.

Figure 7-1 depicts the basic concept of SDN.

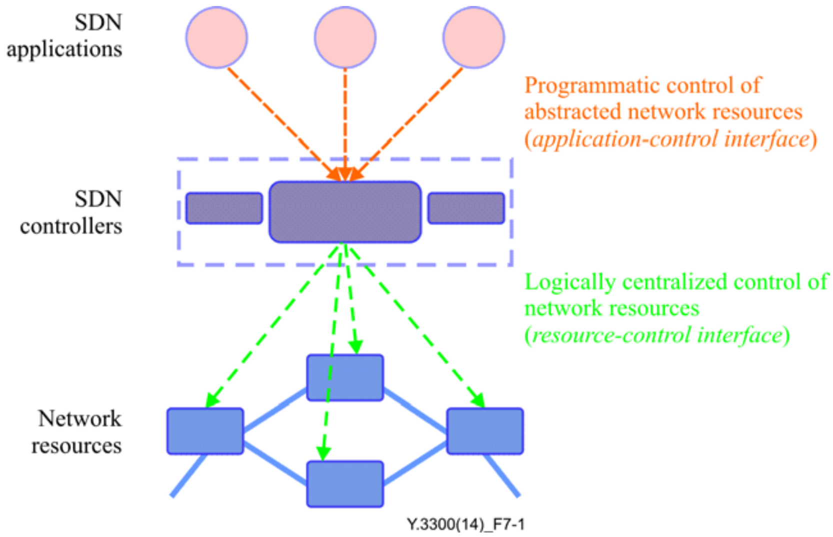

Figure 7-1 – Concept of SDN

SDN relocates the control of network resources to a dedicated network element, namely SDN controller, which provides a means to program, orchestrate, control and manage the network resources through software (i.e., SDN applications).

The distributed network resources perform the network functions such as data packet transport and processing, but the behaviors of network resources are directly controlled via a standardized interface (i.e., *resource-control interface*) and the relevant information and data models. See [b-IETF RFC 3444]. The SDN controller uses the interface and arbitrates the control of network resources in a logically centralized manner.

The SDN controller manages and configures the distributed network resources and provides an abstracted view of the network resources to the SDN applications via another standardized interface (i.e., *application-control interface*) and the relevant information and data models. The SDN application can customize and automate the operations (including management) of the abstracted network resources in a programmable manner via this interface. Note that the SDN controller can provide different types of interfaces to SDN applications (e.g., more abstracted and more object-oriented).

8 Objectives

The objectives of SDN are to realize:

– *Faster network business cycles*

  SDN reduces the response time of business requests to network providers, e.g., to increase customer satisfaction or to shorten the payback time of investment through further automation of network operations;

– *Acceleration of innovation*

  SDN accelerates business and/or technical innovation through more flexibility of the network operations, thus making trials easier;

– *Accelerated adaptation to customer demands*

  SDN facilitates the accommodation of customer's connectivity requirements by means of dynamic negotiation of network service characteristics and of dynamic network resource control;

– *Improved resource availability and usage efficiency*

  SDN is meant to improve network resource availability and efficiency, in particular when combined with network virtualization, due to the introduction of a high level of automation in the overall service delivery and operation procedures, from service parameter negotiation to fulfilment and assurance;

– *Customization of network resources including service-aware networking*

  SDN allows network customization for the network services which have different requirements, through the programming of network resource operations, including the dynamic enforcement of a set of policies (e.g., resource planning as a function of the number of customers' orders to be processed over time, forwarding and routing, quality of service (QoS) and traffic engineering, security).

9 High-level capabilities

SDN provides the following high-level capabilities:

– *Programmability*

  The behaviour of network resources can be customized by SDN applications through a standardized programming interface for network control and management functionality. The user of the interface may be network providers, service providers, and customers including end-users. This enables the SDN applications to automate the operations of network resources according to their needs.

– *Resource abstraction*

  The property and behaviour of underlying network resources can be appropriately abstracted and understood, orchestrated, controlled and/or managed by those who program them, thanks to relevant, standard information and data models. These models provide a detailed, abstracted view of physical or virtualized network resources.

Programmability contributes to the introduction of a high level of automation in the overall service delivery procedure, to achieve business agility, such as dynamic service creation and provisioning. A standardized interface providing a channel for interactions between SDN applications and SDN controllers is used to access network information and to program application-specific network behaviors. This programmability provides the ability to control or configure the network elements by a logically centralized SDN controller through another standardized interface.

NOTE – The delivery of some network services assumes the combination of a variety of functionalities for optimized packet processing and traffic forwarding purposes. Programmability will, in this case, allow to smartly combine these various functionalities into an ordered set of function chains. See [b-IETF SFC]. To support programmability, resource abstraction is a desirable capability. Information and data models are means to provide an abstracted view of the underlying network resources to the SDN applications, so that the application developers can simplify their program logic without the need for detailed knowledge of the underlying network resources and technologies.

10 Requirements

SDN provides programmability of network control and abstraction of underlying network resources, among other SDN features. As described in clause 6, network resources as used in this context of requirements refer to network elements including switches and routers.

In order to support these capabilities, the requirements for SDN are described as follows:

– SDN is required to support programmability of network resources;
– SDN is required to support orchestration of network resources and SDN applications;
– SDN is required to provide an application-control interface for customizing the behaviour of network resources;
– SDN is required to provide a resource-control interface for control of network resources;
– SDN is required to provide logically centralized control of network resources;
– SDN is required to separate SDN control from the network resources;
– SDN is required to support the abstraction of underlying network resources, by means of standard information and data models;
– SDN is required to support the management of physical network resources;
– SDN is recommended to support the management of virtual network resources.

11 High-level architecture

The high-level architecture of SDN consists of several layers as depicted in Figure 11-1.

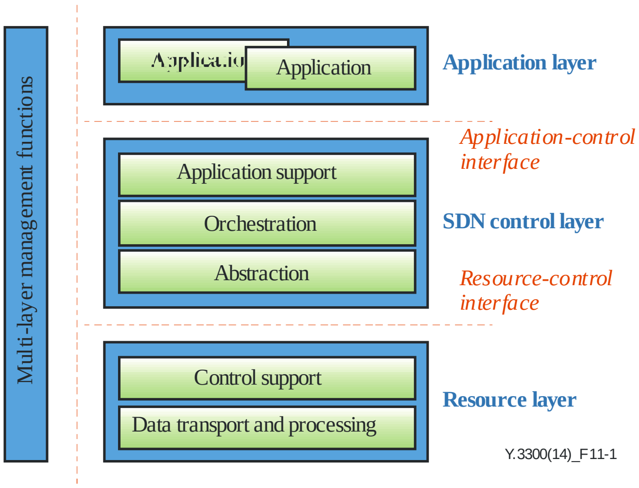

Figure 11-1 – High-level architecture of SDN

11.1 Application layer

The application layer is where SDN applications specify network services or business applications by defining a service-aware behaviour of network resources in a programmatic manner. These applications interact with the SDN control layer via application-control interfaces, in order for the SDN control layer to automatically customize the behaviour and the properties of network resources. The programming of an SDN application makes use of the abstracted view of the network resources provided by the SDN control layer by means of information and data models exposed via the application-control interface.

11.2 SDN control layer

The SDN control layer provides a means to dynamically and deterministically control the behaviour of network resources (such as data transport and processing), as instructed by the application layer. The SDN applications specify how network resources should be controlled and allocated, by interacting with the SDN control layer via application-control interfaces. The control signalling from the SDN control layer to the network resources is then delivered via resource-control interfaces. The configuration and/or properties exposed to SDN applications are abstracted by means of information and data models. The level of abstraction varies according to the applications and the nature of the services to be delivered.

– *Application support*

  The application-support function provides application-control interface for SDN applications to access network information and program application-specific network behaviours.

– *Orchestration*

  The orchestration function provides the automated control and management of network resources and coordination of requests from the application layer for network resources based on the policy provided by the multi-layer management functions or the application layer.

  The orchestration function provides control and management of network resources covering management of physical and virtual network topologies, network elements, and traffic for example. It also interacts with the multi-layer management functions to provide management of SDN application-related operations such as user management, service creation and provisioning.

– *Abstraction*

  The abstraction function interacts with network resources, and provides an abstraction of the network resources, including network capabilities and characteristics, in order to support management and orchestration of physical and virtual network resources. Such abstraction relies upon standard information and data models and is independent of the underlying transport infrastructure.

NOTE – The SDN control layer, instead of the application layer, may execute control for the resource layer for the sake of performance (e.g., for traffic engineering).

11.3 Resource layer

The resource layer is where the network elements perform the transport and the processing of data packets according to the decisions made by the SDN control layer, and which have been forwarded to the resource layer via a resource-control interface.

– *Control support*

  The control support function interacts with the SDN control layer and supports programmability via resource-control interfaces.

– *Data transport and processing*

  The data transport and processing function provides data forwarding and data routing functionalities.

  The data forwarding functionality handles the incoming data flows to forward them along the data forwarding paths that have been computed and established according to the requirements defined by the SDN applications. The control of the data forwarding functionality is provided by the SDN control layer so that the data forwarding functionality in the resource layer may be minimized.

The data routing functionality provides network control and services in the resource layer according to the routing rules which can be customized by the SDN control layer for SDN applications. The data forwarding paths are determined by distributed routing control of the resource layer.

Note that the resource layer may provide either data forwarding functionality or data routing functionality, or both.

Additional functionalities including media transcoding and data compression can be added, removed, or reconfigured for customization of data packets as specified by SDN applications.

NOTE – Additional functionalities including media transcoding and data compression can be invoked for the delivery of a service. From an SDN perspective, the ability to solicit a variety of functions to dynamically structure a service and allocate resources accordingly is a promising opportunity to enforce differentiated traffic forwarding policies within the network, thereby contributing to the improvement of customer's quality of experience.

11.4 Multi-layer management functions

The multi-layer management functions exist as cross-layer functions providing functionalities for managing, as appropriate, the functionalities of the other layers, i.e., the application layer, the SDN control layer and the resource layer.

The multi-layer management functions include functionalities for supporting fault, configuration, accounting, performance and security (FCAPS) management as described in [ITU-T M.3400]. Examples of such functionalities are equipment inventory, software upgrade, fault isolation, performance optimization, energy efficient operations, and initial configuration of the network resources, SDN controllers, and SDN applications. Specifically, autonomic management, i.e., continuous adaptation to the network status, may be performed by the SDN control layer.

11.5 Interfaces

There are two interfaces in the high-level architecture of SDN: the application-control interface and the resource-control interface. They provide access to the SDN controllers and network resources, allowing for programmable control of network resources.

The details of each interface are as follows:

– *Resource-control interface*

The resource-control interface is used for interactions between the SDN control layer and the resource layer. Information exchanged through these interactions include controlling information provided by the SDN control layer to the resource layer (e.g., for configuring a network resource or providing policies) as well as the information that pertains to the (unsolicited) notifications sent by the resource layer whenever the network topology changes, congestion is detected, etc. Such notification information also includes data that are meant to assess the conformance of what has been delivered against what has been negotiated (hence the notion of service fulfilment and assurance). This interface provides high-level accesses to the network resources regardless of their respective technology.

– *Application-control interface*

The application-control interface is used for interactions between the application layer and the SDN control layer. The application-control interface can be used by the application layer to feed the SDN control layer information that will contribute to the decision-making process in terms of dynamic resource allocation or policy enforcement, for example. The SDN control layer can also expose the information and data models that reflect the resource abstraction to the application layer through this interface.

12 Environmental considerations

SDN is meant to facilitate self-adaptability, so that resource availability and usage efficiency can be improved. This is likely to contribute to the optimization of resource usage and therefore reduced energy consumption.

SDN relocates the control of network resources to a logically centralized SDN controller. This may contribute to simplify the design of network elements, thus power consumption is expected to decrease. However, the function that is logically centralized may become complicated, thus its power consumption may increase.

13 Security considerations

The introduction of a high level of automation in the overall service delivery procedure by means of SDN techniques inevitably raises security challenges. In particular, access to network resources by applications must be granted in order to protect the networking infrastructure and its components from a denial of service attack that may jeopardize the overall robustness, quality and reliability of the SDN architecture, or the services that it delivers.

SDN provides new possibilities to combat security breaches. The affected resources may be easily and quickly isolated, malicious traffic may be safely terminated, sensitive flows can be identified and separately transferred in a more secure manner, e.g., with dedicated equipment and security protocols. All these processes may be automated due to SDN for improved availability. Moreover, a logically centralized control of SDN enables operators and/or entities to have a broader and/or a global view of the current status of networks, which makes security operations easier and more efficient.

On the other hand, SDN may aggravate the damage of security breaches, misconfiguration, privacy infringement and other incidents. Properties that were traditionally implemented in hardware and impossible to change can now be modified, misconfigured or can function improperly. Such damage can expand quickly as responses of various software programs and human operation may be too slow for appropriate reaction. It is therefore necessary to enhance monitoring capability and automated operations. More careful checking of e.g., policy configuration, becomes necessary. Moreover, a logically centralized controller can be a single point of failure, and can be a target of malicious attacks, thus special attention is required.

Appendix I

Areas for further considerations in SDN standardization

(This appendix does not form an integral part of this Recommendation.)

In this appendix, technical areas for further considerations in SDN are described.

I.1 Interworking

A network in one administrative domain is usually controlled and managed by the service policy applied by one network provider. However, integrated network services among multiple administrative domains need to be composed of relevant network services in each administrative domain. It is important that SDN provides interworking functionality for such integrated network services among multiple administrative domains taking into account the following points:

– It is important that SDN exchange available network parameters (e.g., bandwidth, latency, label or id information) for automating the control and/or management of the network services among different administrative domains. These network parameters may be screened or abstracted because such network parameters contain sensitive information in many cases. This functionality may also need to configure a part of other administrative domain networks and get statistics about the domain.

The following points are not specific to SDN, but need consideration.

– In forwarding packets across multiple administrative domains, packet formats may be different, and it is important to support converting the format of packets and/or adjusting the network properties (e.g., bandwidth and latency) before or after packets enter another network;

– It is also important to support authentication, authorization, and accounting (AAA) features because interworking operations usually need computational resources from other administrative domains.

I.2 Verification of SDN applications

With SDN technologies, network providers, service providers, and customers can customize the network resources by writing an SDN application. However, incomplete or malicious SDN applications could cause a breakdown of underlying networks shared by heterogeneous network elements and stakeholders. Thus, it is desirable that SDN applications be formally specified and verified (i.e., formal method) to prevent them from misinterpreting their intentions on network operations and to avoid inconsistency within the network.

Formal methods are software engineering techniques based on mathematical representation and analysis of software programs [b-FM Clarke]. Such formal methods can be used for analysis of specification and verification of software behaviour.

Using the formal methods for SDN applications can minimize the risk of misuse or malfunction of SDN applications because the formal method serves to remove ambiguity, inconsistency, and possible conflicts of SDN applications in the networks. Furthermore, network operators can use the formal methods to check consistency and safety of their network configurations, virtual/physical topologies and networking resources. This enables confirmation of their configurations and behaviours of various networking properties. Some examples of these properties are:

– no routing loops or unreachable points in the network;

– no rule or behaviour conflicts between multiple SDN applications;

– no conflicts on physical network resource assignments between different SDN applications;

– no conflicts in dynamic network update where new or updated network configurations conform to properties of the network and do not break consistency of existing networks.

I.3 Adaptation to large-scale networks

SDN should provide connectivity from a local area to a wide area, as current networks do. Because SDN provides logically centralized control of networks, adaptation to large-scale networks is important. A large-scale network has two aspects: a single SDN domain composed of many network elements, and multiple SDN domains managed by different network providers. While SDN may support many network elements in a wide area, the number of network elements that a single SDN controller can accommodate is limited. In this regard, scalability needs to be considered. One way to provide scalability is to have SDN controllers logically centralized but physically distributed, to correspond to a wide area.

In large-scale networks, reliability is a particularly important issue. As a characteristic of logically centralized control in SDN, an SDN controller tends to become a single point of failure. The SDN controller may be replicated to improve reliability.

I.4 Design of resource abstraction

To make programmability of SDN easier and more efficient, it is important that resource abstraction is appropriately designed and adopted for the best practice and performance of networks. It is important that resource abstraction be provided with the following points:

– *Common resource abstraction model*

  It is important that a common resource abstraction model be applied to similar network resources regardless of the underlying technology, since a common information model simplifies the programming for the resources. Widely applicable standard information and data models require the appropriate abstraction of various capabilities, such as packet forwarding, wireless interfaces, and path computation-related information (e.g., routing metrics).

– *Appropriate granularity of abstraction*

  It is not appropriate to show all network resources and their characteristics to SDN applications. On one hand, not all the details and characteristics of the network resources are necessary for SDN application programming. On the other hand, excessive abstraction prevents SDN applications from utilizing network resources to the maximum because it hides the details of the resources. It is therefore important that SDN provides an appropriate and balanced granularity of abstraction.

– *Performance tuning of network resources*

  When network resources are abstracted, an SDN application may not be able to access some network resource-specific parameters, and this may degrade the performance. Suitable abstraction or some mechanisms to tune the performance of network resources automatically are necessary.

I.5 Virtualization of network elements

Network elements, which can be abstracted resources, may be shared among multiple applications. But each SDN application does the programming of network elements according to the SDN application requirements. In that case, it is necessary that these networks and network elements are mutually isolated. Toward this end, network elements should be virtually partitioned to support each application network.

SDN applications may require network resources, e.g., bandwidth and packet processing, whereas the applications' requirements cannot be satisfied with a single network element. In that case, multiple network elements can be virtually combined to appear as a single resource to the applications to satisfy the requirements without further management efforts of multiple network resources.

For the above two cases, virtualization techniques can provide an appropriate solution because they assume network element partition and aggregation, as well as a single interface to such virtualized network resources.

I.6 Multiple-level of programmability

In conventional networks, each resource in multiple open system interconnection (OSI) layers [b-ITU-T X.200] is managed by different management entities. This makes it difficult to orchestrate total performance and to recover from network failures quickly. SDN is desirable to provide control and management programming interfaces to network resources, e.g., for coordination and status monitoring purposes. In addition, the demands of SDN applications can be diverse and may be related to multiple OSI layers.

The interfaces need to span multiple OSI layers and work in a unified manner: multiple-level of programmability that spans L1-L7 OSI layers is an important issue to be addressed for SDN.

One example for such multi-level programmability is cut-through optical path computation. In this example, some SDN components may dynamically change the target OSI layer to allocate a path, depending on traffic characteristics. For example, SDN components may first allocate a path in the OSI layer 3, namely an IP route. After checking OSI layer 3 header and its applicability, SDN intelligence may then decide instead to allocate an optical path, e.g., to provide additional bandwidth.

I.7 Programmatic extension in resource layer

In order to improve network programmability, it is desirable to extend the functions of the resource layer on demand in a programmable manner [b-SDN-WS Nakao]. This ability can dynamically add or remove additional functions for data transport and processing (such as packet caching, header/payload compression, regular expression matching, data transcoding, or even handling newly developed protocols) as per SDN application requirements or can dynamically update the control support function, thereby avoiding hardware replacement.

Thus, rapid development, deployment, and replacement of resource layer functionalities and resource-control interfaces leads to timely and tailored service solutions aligned with requirements of applications and operator policies.

I.8 Management

The following management functionalities specific to one of the layers need to be further considered:

– Management of the underlying physical and virtual network resources, providing support for programmatically extendable SDN data transport and processing functions, which is described in clause I.7;

– Management of the software and hardware platform of an SDN controller which includes lifecycle management of dynamically upgradable functions of the SDN controller and FCAPS.

The multi-layer management functions should interoperate with 3rd party management functions, for example, for billing, customer care, statistics collection or dynamic service provisioning.

Another issue to be considered is how the multi-layer management functions are deployed. Possible approaches include the use of a centralized management model or a hybrid model. In the hybrid model, some management functions are distributed, while others are implemented in centralized management systems. With such an approach, network resources support management functions that may improve network robustness and scalability as well as shorten the management system response time. In both models, there is an SDN management system that performs all centralized management operations.

Bibliography

[b-ITU-T X.200] Recommendation ITU-T X.200 (1994), *Information technology – Open Systems Interconnection – Basic Reference Model: The basic model.*

[b-ITU-T Y.2622] Recommendation ITU-T Y.2622 (2012), *Architecture of an independent scalable control plane in future packet based networks*.

[b-ETSI NFV] ETSI NFV ISG, *Network Functions Virtualisation*, http://portal.etsi.org/portal/server.pt/community/NFV

[b-IETF I2RS] IETF I2RS Working Group, *Interface to the Routing System*, http://datatracker.ietf.org/wg/i2rs/charter/

[b-IETF RFC 3444] IETF RFC 3444 (2003), *On the Difference between Information Models and Data Models*.

[b-IETF RFC 3746] IETF RFC 3746 (2004), *Forwarding and Control Element Separation (ForCES) Framework*.

[b-IETF RFC 7149] IETF RFC 7149 (2014), *Software-Defined Networking: A Perspective from within a Service Provider Environment*.

[b-IETF SFC] IETF Service Function Chaining (sfc) working group, http://datatracker.ietf.org/wg/sfc/charter/

[b-ONF] Open Networking Foundation, "OpenFlow/Software-Defined Networking (SDN)," https://www.opennetworking.org/.

[b-OpenDayLight] Linux Foundation, OpenDayLight, http://www.opendaylight.org/.

[b-SDN-WS Nakao] "*Deeply Programmable Network*", Emerging Technologies for Network Virtualization, and Software Defined Network (SDN), ITU-T Workshop on Software Defined Networking (SDN), http://www.itu.int/en/ITU-T/Workshops-and-Seminars/sdn/201306/Documents/KS-Aki_Nako_rev1.pdf.

[b-FM Clarke] Edmund M. Clarke and Jeannette M. Wing (1996), *Formal methods: State of the art and future directions*, ACM Computing Surveys Vol. 28, No. 4, December; pp. 626-643.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	General tariff principles
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Terminals and subjective and objective assessment methods
Series Q	Switching and signalling
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	**Global information infrastructure, Internet protocol aspects and next-generation networks**
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2014