

SDN Research Group  
Internet-Draft  
Intended status: Standards Track  
Expires: May 4, 2017

LM. Contreras  
Telefonica  
CJ. Bernardos  
UC3M  
D. Lopez  
Telefonica  
M. Boucadair  
Orange  
P. Iovanna  
Ericsson  
October 31, 2016

**Cooperating Layered Architecture for SDN  
draft-irtf-sdnrg-layered-sdn-01**

Abstract

Software Defined Networking proposes the separation of the control plane from the data plane in the network nodes and its logical centralization on a control entity. Most of the network intelligence is moved to this functional entity. Typically, such entity is seen as a compendium of interacting control functions in a vertical, tight integrated fashion. The relocation of the control functions from a number of distributed network nodes to a logical central entity conceptually places together a number of control capabilities with different purposes. As a consequence, the existing solutions do not provide a clear separation between transport control and services that relies upon transport capabilities.

This document describes a proposal named Cooperating Layered Architecture for SDN. The idea behind that is to differentiate the control functions associated to transport from those related to services, in such a way that they can be provided and maintained independently, and can follow their own evolution path.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of [BCP 78](#) and [BCP 79](#).

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at <http://datatracker.ietf.org/drafts/current/>.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any

time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on May 4, 2017.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to [BCP 78](#) and the IETF Trust's Legal Provisions Relating to IETF Documents (<http://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

Table of Contents

- [1.](#) Introduction . . . . . [3](#)
- [2.](#) Terminology . . . . . [4](#)
- [3.](#) Architecture overview . . . . . [5](#)
  - [3.1.](#) Functional strata . . . . . [8](#)
    - [3.1.1.](#) Connectivity stratum . . . . . [8](#)
    - [3.1.2.](#) Service stratum . . . . . [9](#)
    - [3.1.3.](#) Recursiveness . . . . . [9](#)
  - [3.2.](#) Plane separation . . . . . [9](#)
    - [3.2.1.](#) Control Plane . . . . . [9](#)
    - [3.2.2.](#) Management Plane . . . . . [10](#)
    - [3.2.3.](#) Resource Plane . . . . . [10](#)
- [4.](#) Required features . . . . . [10](#)
- [5.](#) Communication between SDN Controllers . . . . . [11](#)
- [6.](#) Deployment scenarios . . . . . [11](#)
  - [6.1.](#) Full SDN environments . . . . . [11](#)
    - 6.1.1. Multiple Service strata associated to a single Connectivity stratum . . . . . [11](#)
    - 6.1.2. Single service stratum associated to multiple Connectivity strata . . . . . [12](#)
  - [6.2.](#) Hybrid environments . . . . . [12](#)
    - 6.2.1. SDN Service stratum associated to a legacy Connectivity stratum . . . . . [12](#)
    - 6.2.2. Legacy Service stratum associated to an SDN Connectivity stratum . . . . . [12](#)
  - [6.3.](#) Multi-domain scenarios in Connectivity Stratum . . . . . [12](#)
- [7.](#) Use cases . . . . . [13](#)



[7.1. Network Function Virtualization . . . . .](#) [13](#)  
[7.2. Abstraction and Control of Transport Networks . . . . .](#) [13](#)  
[8. IANA Considerations . . . . .](#) [13](#)  
[9. Security Considerations . . . . .](#) [13](#)  
[10. References . . . . .](#) [13](#)  
[10.1. Normative References . . . . .](#) [13](#)  
[10.2. Informative References . . . . .](#) [14](#)  
 Authors' Addresses . . . . . [14](#)

**1. Introduction**

Software Defined Networking (SDN) proposes the separation of the control plane from the data plane in the network nodes and its logical centralization on a control entity. A programmatic interface is defined between such entity and the network nodes, which functionality is supposed to perform traffic forwarding (only). Through that interface, the control entity instructs the nodes involved in the forwarding plane and modifies their traffic forwarding behavior accordingly.

Most of the intelligence is moved to such functional entity. Typically, such entity is seen as a compendium of interacting control functions in a vertical, tight integrated fashion.

This approach presents a number of issues:

- o Unclear responsibilities between actors involved in a service provision and delivery.
- o Complex reuse of functions for the provision of services.
- o Closed, monolithic control architectures.
- o Difficult interoperability and interchangeability of functional components.
- o Blurred business boundaries among providers.
- o Complex service/network diagnosis and troubleshooting, particularly to determine which segment is responsible for a failure.

The relocation of the control functions from a number of distributed network nodes to another entity conceptually places together a number of control capabilities with different purposes. As a consequence, the existing solutions do not provide a clear separation between services and transport control.



This document describes a proposal named Cooperating Layered Architecture for SDN (CLAS). The idea behind that is to differentiate the control functions associated to transport from those related to services, in such a way that they can be provided and maintained independently, and can follow their own evolution path.

Despite such differentiation it is required a close cooperation between service and transport layers and associated components to provide an efficient usage of the resources.

## 2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119](#) [[RFC2119](#)].

This document makes use of the following terms:

- o Transport: denotes the transfer capabilities offered by a networking infrastructure. The transfer capabilities can rely upon pure IP techniques, or other means such as MPLS or optics.
- o Service: denote a logical construct that make use of transport capabilities. This document does not make any assumption on the functional perimeter of a service that can be built above a transport infrastructure. As such, a service can be an offering that is offered to customers or be invoked for the delivery of another (added-value) service.
- o SDN intelligence: refers to the decision-making process that is hosted by a node or a set of nodes. The intelligence can be centralized or distributed. Both schemes are within the scope of this document. The SDN intelligence relies on inputs form various functional blocks such as: network topology discovery, service topology discovery, resource allocation, business guidelines, customer profiles, service profiles, etc. The exact decomposition of an SDN intelligence, apart from the layering discussed in this document, is out of scope.

Additionally, the following acronyms are used in this document.

CLAS: Cooperating Layered Architecture for SDN

FCAPS: Fault, Configuration, Accounting, Performance and Security

SDN: Software Defined Networking



SLA: Service Level Agreement

### 3. Architecture overview

Current operator networks support multiple services (e.g., VoIP, IPTV, mobile VoIP, critical mission applications, etc.) on a variety of transport technologies. The provision and delivery of a service independently of the underlying transport capabilities requires a separation of the service related functionalities and an abstraction of the transport network to hide the specificities of underlying transfer techniques while offering a common set of capabilities.

Such separation can provide configuration flexibility and adaptability from the point of view of either the services or the transport network. Multiple services can be provided on top of a common transport infrastructure, and similarly, different technologies can accommodate the connectivity requirements of a certain service. A close coordination among them is required for a consistent service delivery (inter-layer cooperation).

This document focuses particularly on:

- o Means to expose transport capabilities to external services.
- o Means to capture service requirements of services.
- o Means to notify service intelligence with underlying transport events, for example to adjust service decision-making process with underlying transport events.
- o Means to instruct the underlying transport capabilities to accommodate new requirements, etc.

An example is to guarantee some Quality of Service (QoS) levels. Different QoS-based offerings could be present at both service and transport layers. Vertical mechanisms for linking both service and transport QoS mechanisms should be in place to provide the quality guarantees to the end user.

CLAS architecture assumes that the logically centralized control functions are separated in two functional blocks or layers. One of the functional blocks comprises the service-related functions, whereas the other one contains the transport-related functions. The cooperation between the two layers is considered to be implemented through standard interfaces.

Figure 1 shows the CLAS architecture. It is based on functional separation in the NGN architecture defined by the ITU-T in [[Y.2011](#)].





Two strata of functionality are defined, namely the Service Stratum, comprising the service-related functions, and the Connectivity Stratum, covering the transport ones. The functions on each of these layers are further grouped on control, management and user (or data) planes.

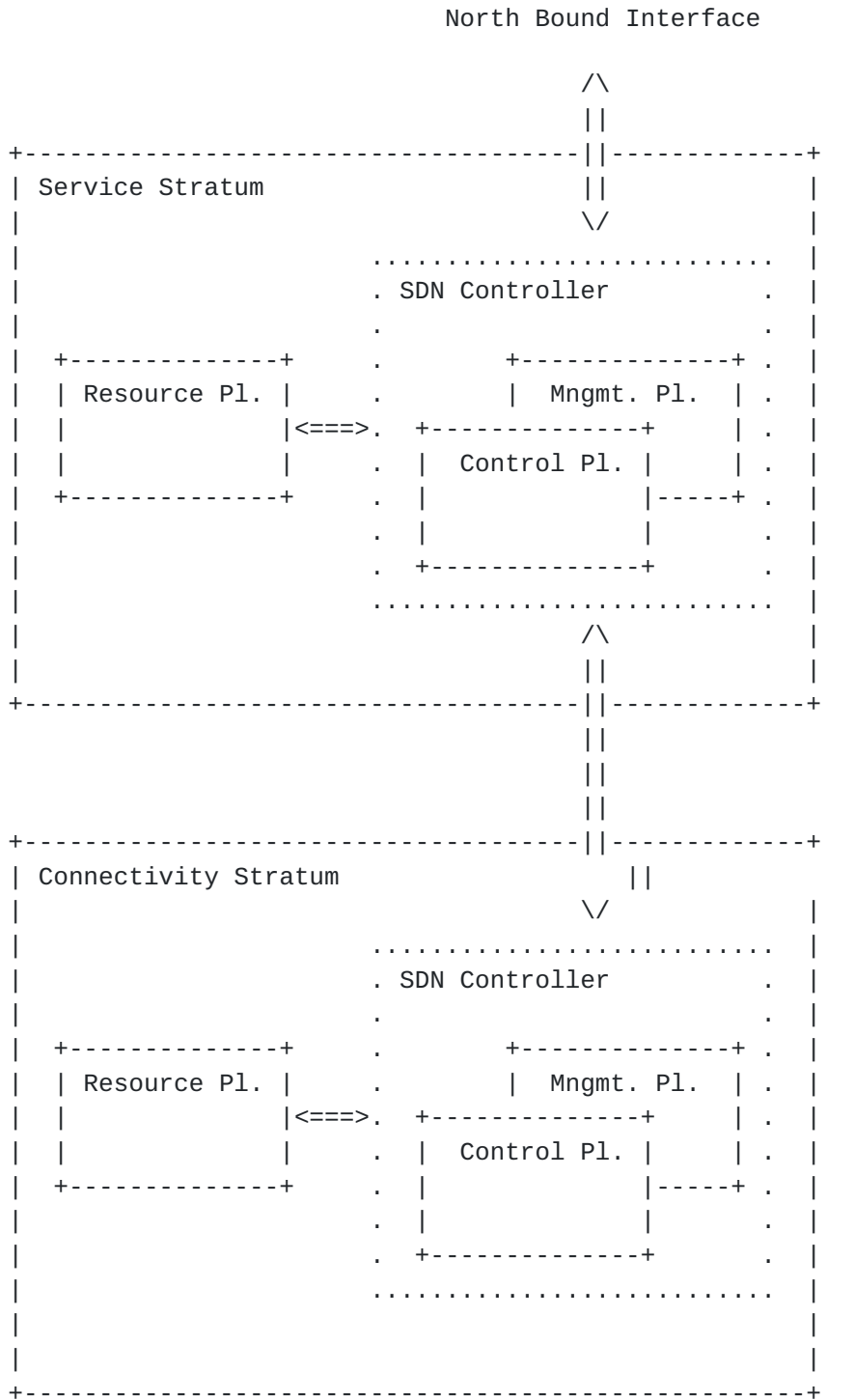


Figure 1: Cooperating Layered Architecture for SDN

In the CLAS architecture both the control and management functions are the ones logically centralized in one or a set of SDN controllers, in such a way that separated SDN controllers are present



in the Service and Connectivity strata. Furthermore, the generic user or data plane functions included in the NGN architecture are referred here as resource plane functions. The resource plane in each stratum is controlled by the corresponding SDN controller through a standard interface.

The SDN controllers cooperate for the provision and delivery of services. There is a hierarchy in which the Service SDN controller requests transport capabilities to the Transport SDN controller. Furthermore, the Transport SDN controller interacts with the Service SDN controller to inform it about events in the transport network that can motivate actions in the service layer.

The Service SDN controller acts as a client of the Transport SDN controller.

Despite it is not shown in the figure, the Resource planes of each stratum could be connected. This will depend on the kind of service provided. Furthermore, the Service stratum could offer an interface towards external applications to expose network service capabilities to those applications or customers.

This document does assume that SDN techniques can be enabled jointly with other distributed means (e.g., IGP).

### **3.1. Functional strata**

As described before, the functional split separates transport-related functions from service-related functions. Both strata cooperate for a consistent service delivery.

Consistency is determined and characterized by the service layer.

Communication between these two components could be implemented using a variety of means (such as [[I-D.boucadair-connectivity-provisioning-protocol](#)], Intermediate-Controller Plane Interface (I-CPI) [[ONFArch](#)], etc).

#### **3.1.1. Connectivity stratum**

The Connectivity stratum comprises the functions focused on the transfer of data between the communication end points (e.g., between end-user devices, between two service gateways, etc.). The data forwarding nodes are controlled and managed by the Transport SDN component. The Control plane in the SDN controller is in charge of instructing the forwarding devices to build the end to end data path for each communication or to make sure forwarding service is appropriately setup. Forwarding may not be rely on the sole pre-



configured entries; dynamic means can be enabled so that involved nodes can build dynamically routing and forwarding paths. Finally, the Management plane performs management functions (i.e., FCAPS) on those devices, like fault or performance management, as part of the Connectivity stratum capabilities.

### **3.1.2. Service stratum**

The Service stratum contains the functions related to the provision of services and the capabilities offered to external applications. The Resource plane consists of the resources involved in the service delivery, such as computing resources, registries, databases, etc. The Control plane is in charge of controlling and configuring those resources, as well as interacting with the Control plane of the Transport stratum in client mode for requesting transport capabilities for a given service. In the same way, the Management plane implements management actions on the service-related resources and interacts with the Management plane in the Connectivity stratum for a cooperating management between layers.

### **3.1.3. Recursiveness**

Recursive layering can happen in some usage scenarios in which the Connectivity Stratum is itself structured in Service and Connectivity Stratum. This could be the case of the provision of a transport services complemented with advanced capabilities additional to the pure data transport (e.g., maintenance of a given SLA [[RFC7297](#)]).

## **3.2. Plane separation**

The CLAS architecture leverages on the SDN proposition of plane separation. As mentioned before, three different planes are considered for each stratum. The communication among these three planes (and with the corresponding plane in other strata) is based on open, standard interfaces.

### **3.2.1. Control Plane**

The Control plane logically centralizes the control functions of each stratum and directly controls the corresponding resources. [[RFC7426](#)] introduces the role of the control plane in a SDN architecture. This plane is part of an SDN controller, and can interact with other control planes in the same or different strata for accomplishing control functions.





### **3.2.2. Management Plane**

The Management plane logically centralizes the management functions for each stratum, including the management of the Control and Resource planes. [[RFC7426](#)] describes the functions of the management plane in a SDN environment. This plane is also part of the SDN controller, and can interact with the corresponding management planes residing in SDN controllers of the same or different strata.

### **3.2.3. Resource Plane**

The Resource plane comprises the resources for either the transport or the service functions. In some cases the service resources can be connected to the transport ones (e.g., being the terminating points of a transport function) whereas in other cases it can be decoupled from the transport resources (e.g., one database keeping some register for the end user). Both forwarding and operational planes proposed in [[RFC7426](#)] would be part of the Resource plane in this architecture.

## **4. Required features**

A number of features are required to be supported by the CLAS architecture.

- o Abstraction: the mapping of physical resources into the corresponding abstracted resources.
- o Service parameter translation: translation of service parameters (e.g., in the form of SLAs) to transport parameters (or capabilities) according to different policies.
- o Monitoring: mechanisms (e.g. event notifications) available in order to dynamically update the (abstracted) resources' status taking in to account e.g. the traffic load.
- o Resource computation: functions able to decide which resources will be used for a given service request. As an example, functions like PCE could be used to compute/select/decide a certain path.
- o Orchestration: ability to combine diverse resources (e.g., IT and network resources) in an optimal way.
- o Accounting: record of resource usage.
- o Security: secure communication among components, preventing e.g. DoS attacks.



## **5. Communication between SDN Controllers**

The SDN Controller residing respectively in the Service and the Connectivity Stratum need to establish a tight coordination. Mechanisms for transfer relevant information for each stratum should be defined.

From the Service perspective, the Service SDN controller needs to easily access transport resources through well defined APIs to access the capabilities offered by the Connectivity Stratum. There could be different ways of obtainign such transport-aware information, i.e., by discovering or publishing mechanisms. In the former case the Service SDN Controller could be able of handling complete information about the transport capabilities (including resources) offered by the Connectivity Stratum. In the latter case, the Connectivity Stratum exposes available capabilities e.g. through a catalog, reducing the amount of detail of the underlying network.

On the other hand, the Connectivity Stratum requires to properly capture Service requirements. These can include SLA requirements with specific metrics (such as delay), level of protection to be provided, max/min capacity, applicable resource constraints, etc.

The communication between controllers should be also secure, preventing denial of service.

## **6. Deployment scenarios**

Different situations can be found depending on the characteristics of the networks involved in a given deployment.

### **6.1. Full SDN environments**

This case considers the fact that the networks involved in the provision and delivery of a given service have SDN capabilities.

#### **6.1.1. Multiple Service strata associated to a single Connectivity stratum**

A single Connectivity stratum can provide transfer functions to more than one Service strata. The Connectivity stratum offers a standard interface to each of the Service strata. The Service strata are the clients of the Connectivity stratum. Some of the capabilities offered by the Connectivity stratum can be isolation of the transport resources (slicing), independent routing, etc.



### **6.1.2. Single service stratum associated to multiple Connectivity strata**

A single Service stratum can make use of different Connectivity strata for the provision of a certain service. The Service stratum interfaces each of the Connectivity strata with standard protocols, and orchestrates the provided transfer capabilities for building the end to end transport needs.

## **6.2. Hybrid environments**

This case considers scenarios where one of the strata is legacy totally or in part.

### **6.2.1. SDN Service stratum associated to a legacy Connectivity stratum**

An SDN service stratum can interact with a legacy Connectivity stratum through some interworking function able to adapt SDN-based control and management service-related commands to legacy transport-related protocols, as expected by the legacy Connectivity stratum. The SDN controller in the Service stratum is not aware of the legacy nature of the underlying Connectivity stratum.

### **6.2.2. Legacy Service stratum associated to an SDN Connectivity stratum**

A legacy Service stratum can work with an SDN-enabled Connectivity stratum through the mediation of an interworking function capable to interpret commands from the legacy service functions and translate them into SDN protocols for operating with the SDN-enabled Connectivity stratum.

## **6.3. Multi-domain scenarios in Connectivity Stratum**

The Connectivity Stratum can be composed by transport resources being part of different administrative, topological or technological domains. The Service Stratum can yet interact with a single entity in the Connectivity Stratum in case some abstraction capabilities are provided in the transport part to emulate a single stratum.

Those abstraction capabilities constitute a service itself offered by the Connectivity Stratum to the services making use of it. This service is focused on the provision of transport capabilities, then different of the final communication service using such capabilities.

In this particular case this recursion allows multi-domain scenarios at transport level.



Multi-domain situations can happen in both single-operator and multi-operator scenarios. Multi-operator scenarios will be addressed in future versions of the document.

In single operator scenarios a multi-domain or end-to-end abstraction component can provide an homogeneous abstract view of the underlying heterogeneous transport capabilities for all the domains.

## **7. Use cases**

This section presents a number of use cases as examples of applicability of this proposal

### **7.1. Network Function Virtualization**

NFV environments offer two possible levels of SDN control [[ETSI NFV EVE005](#)]. One level is the need for controlling the NFVI to provide connectivity end-to-end among VNFs (Virtual Network Functions) or among VNFs and PNFs (Physical Network Functions). A second level is the control and configuration of the VNFs themselves (in other words, the configuration of the network service implemented by those VNFs), taking profit of the programmability brought by SDN. Both control concerns are separated in nature. However, interaction between both could be expected in order to optimize, scale or influence each other.

### **7.2. Abstraction and Control of Transport Networks**

To be completed.

## **8. IANA Considerations**

TBD.

## **9. Security Considerations**

TBD. Security in the communication between strata to be addressed.

## **10. References**

### **10.1. Normative References**

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", [BCP 14](#), [RFC 2119](#), DOI 10.17487/RFC2119, March 1997, <<http://www.rfc-editor.org/info/rfc2119>>.





- [Y.2011] "General principles and general reference model for Next Generation Networks", ITU-T Recommendation Y.2011 , October 2004.

## **10.2. Informative References**

- [ETSI\_NFV\_EVE005]  
"Report on SDN Usage in NFV Architectural Framework",  
Dicember 2015.
- [I-D.boucadair-connectivity-provisioning-protocol]  
Boucadair, M., Jacquenet, C., Zhang, D., and P.  
Georgatsos, "Connectivity Provisioning Negotiation  
Protocol (CPNP)", [draft-boucadair-connectivity-  
provisioning-protocol-12](#) (work in progress), June 2016.
- [ONFArch] Open Networking Foundation, "SDN Architecture, Issue 1",  
June 2014,  
<[https://www.opennetworking.org/images/stories/downloads/  
sdn-resources/technical-reports/  
TR\\_SDN\\_ARCH\\_1.0\\_06062014.pdf](https://www.opennetworking.org/images/stories/downloads/sdn-resources/technical-reports/TR_SDN_ARCH_1.0_06062014.pdf)>.
- [RFC7297] Boucadair, M., Jacquenet, C., and N. Wang, "IP  
Connectivity Provisioning Profile (CPP)", [RFC 7297](#),  
DOI 10.17487/RFC7297, July 2014,  
<<http://www.rfc-editor.org/info/rfc7297>>.
- [RFC7426] Haleplidis, E., Ed., Pentikousis, K., Ed., Denazis, S.,  
Hadi Salim, J., Meyer, D., and O. Koufopavlou, "Software-  
Defined Networking (SDN): Layers and Architecture  
Terminology", [RFC 7426](#), DOI 10.17487/RFC7426, January  
2015, <<http://www.rfc-editor.org/info/rfc7426>>.

### Authors' Addresses

Luis M. Contreras  
Telefonica  
Ronda de la Comunicacion, s/n  
Sur-3 building, 3rd floor  
Madrid 28050  
Spain

Email: [luismiguel.contrerasmurillo@telefonica.com](mailto:luismiguel.contrerasmurillo@telefonica.com)  
URI: <http://lmcontreras.com>



Carlos J. Bernardos  
Universidad Carlos III de Madrid  
Av. Universidad, 30  
Leganes, Madrid 28911  
Spain

Phone: +34 91624 6236  
Email: [cjbc@it.uc3m.es](mailto:cjbc@it.uc3m.es)  
URI: <http://www.it.uc3m.es/cjbc/>

Diego R. Lopez  
Telefonica  
Ronda de la Comunicacion, s/n  
Sur-3 building, 3rd floor  
Madrid 28050  
Spain

Email: [diego.r.lopez@telefonica.com](mailto:diego.r.lopez@telefonica.com)

Mohamed Boucadair  
Orange  
Rennes 35000  
France

Email: [mohamed.boucadair@orange.com](mailto:mohamed.boucadair@orange.com)

Paola Iovanna  
Ericsson  
Pisa  
Italy

Email: [paola.iovanna@ericsson.com](mailto:paola.iovanna@ericsson.com)

