An Evolution of Cooperating Layered Architecture for SDN (CLAS) for Compute and Data Awareness

draft-contreras-coinrg-clas-evolution-00

L.M. Contreras (Telefonica), M. Boucadair (Orange),
D. Lopez (Telefonica), C.J. Bernardos (UC3M)
Background

- Cooperating Layered Architecture for Software-Defined Networking (CLAS) was a work adopted in SDNRG which was moved into ISE after RG closure.
- It was finally released as RFC 8597.
- It proposes a layered control architecture where control functions associated with transport are differentiated from those related to services in such a way that they can be provided and maintained independently and can follow their own evolution path.
Overview

• Functional Strata
  – **Service stratum**: functions related to the provision of services (including capabilities exposed to external applications)
  – **Transport stratum**: functions related to the transfer of data between communication endpoints

• Plane separation
  – **Control plane**: control of resources in each strata
  – **Management plane**: management of resources and control plane in each strata
  – **Resource plane**: resources required for a given service (can be or not the termination points of a transport function)

• Despite differentiation, tight cooperation is needed for an efficient service provision
Motivation for CLAS evolution

• Networks are evolving towards a tighter integration of interconnected compute environments
  • Interworking of virtualized and physical service functions

• Moreover, network operations are complementing the capabilities of automation and programmability with the introduction of Artificial Intelligence (AI) and Machine Learning (ML) techniques
  • Base for closed loop automation
Evolved Architecture

Service Stratum
- Resource Plane
- Control Plane
- Mngment Plane

Connectivity Stratum
- Resource Plane
- Control Plane
- Mngment Plane

Compute Stratum
- Resource Plane
- Control Plane
- Mngment Plane

Learning Plane
Augmentation of CLAS with Compute and Data Awareness

• Compute Stratum
  • Consideration of distributed computing capabilities attached to different points in the network, intended for hosting a variety of services and applications usually in a virtualized manner
  • Contains the control, management and resource planes related to the computing part

• Learning Plane
  • Collection, processing and sharing of relevant data from each of the strata.
  • Introduction of Artificial Intelligence (AI) and Machine Learning (ML) techniques in order to improve operations by means of closed loop automation
Potential research directions

• Identify use cases that can help to better define the architecture capabilities
• Work on aspects such as:
  • Communication means/interfaces between strata (and planes)
  • Deployment scenarios (including legacy ones)
  • Potential use cases
  • Link with on-going activities in NMRG (IBN, AI, etc)
• Explore novel architectural approaches: e.g., bus architecture for interaction of planes in a single stratum
• Inter-domain APIs between different/same strata
  • e.g., further developing and updating ideas as described in draft-bernardos-nmrg-multidomain-01
• Explore intent-based APIs/approaches for learning plane
• Data models (and even ontologies) for the exchange and aggregation of information, knowledge and actions among the different planes and strata
Next steps

• Set the scope of the draft aligned with the scope of COINRG

• Collect feedback / interest from the RG on any of the aspects commented

• Prepare a new (more detailed version) for IETF 117