Joint Intra- and Inter-Datacenter Network Optimization and Orchestration

G. Landi[1], M. Capitani[1], A. Kretsis[2], P. Kokkinos[2], K. Christodoulopoulos[2], E. Varvarigos[2]
1: Nextworks, Pisa, Italy
2: Dept. of Computer Engineering and Informatics, University of Patras, Greece

Abstract: We present a hierarchical orchestration platform for inter-domain datacenter networks that includes hybrid optical-electrical intra-datacenter networking and inter-datacenter networking utilizing elastic technologies. We demonstrate dynamic and joint allocation of capacity in an emulated testbed.
OCIS codes: (060.4250) Networks, (060.4256) Networks, network optimization, (060.4251) Networks, assignment and routing algorithms

1. Introduction

Today, hyperscale datacenters (DC) are being built around the world, while dedicated/private wide area transport infrastructures are also deployed for their interconnection. As reported in [1] by 2020 more than 90% of DC traffic will be cloud-DC traffic, originated or terminated in a DC. In addition, in 2020 the traffic between DCs is expected to account for almost 9% of total DC traffic, higher than the 7% reported at the end of 2015, growing faster than the traffic to end users or the traffic within the DC.

Several communication use cases are inter-domain, meaning that they span across multiple DCs, from a server in one DC to another in a different DC, crossing over the wide area transport networks that interconnect the DCs. Examples of such use cases include load balancing, redundancy and disaster recovery, as well as services deployed across a distributed DC, where the virtual machines running at different locations need to interact with some quality of service (QoS) guarantees [2]. Such scenarios require low-latency inter-domain connections, dynamic and efficient capacity allocation and isolation between traffic flows.

A number of technologies promise to satisfy these inter-domain network requirements. In the transport network segment, elastic optical networks EON [3] address the inefficiency problems of traditional WDM networks, providing a fine granular solution for sub- and super-wavelength capacity. Optical technologies for (intra-domain) DC networks (DCN), ranging from hybrid electrical/optical switching to all-optical packet switching have also emerged. NEPHELE EU project [5] leverages hybrid electrical/optical switching to attain the ideal combination of high capacity at reduced cost and power, compared to state-of-the art DC networks.

The orchestration of involved networking technologies and the optimization of their parameters is key to the success and efficiency of inter-domain communications. In this work, we present a Software Defined Networking (SDN) -based optimization and orchestration platform for inter-domain DC networks that include hybrid optical-electrical intra-DC networks and EON inter-DC networks. Using SDN, the forwarding decisions are taken centrally, enabling the on demand adaptation of connections' and the rerouting of traffic according to network and application characteristics. We evaluate the developed system in a realistic testbed and demonstrate its ability to provision guaranteed end-to-end, inter-domain connections.

2. Inter-domain orchestration

The overall architecture for the control of inter-domain network connections between servers located in different datacenters (DC) is presented in Figure 1.

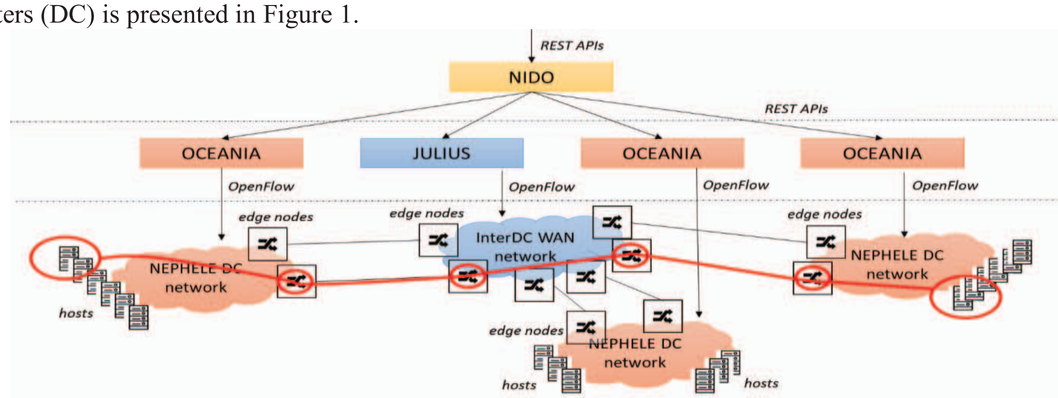


Figure 1: NIDO inter-domain orchestration.

We adopt a hierarchical approach, where a centralized inter-domain network orchestrator, called NEPHELE Inter-Domain network Orchestrator (NIDO) coordinates the action of lower layer intra-domain SDN controllers, namely OCEANIA and JULIUS. The OCEANIA and JULIUS SDN controllers are responsible for network resource allocation in their single domain, (intra-) DC network (DCN) and inter-DC transit domain, respectively. By coordinating these controllers, NIDO achieves end-to-end network resources allocation. This hierarchical approach enables to change the intra-domain controller and/or the related network technology of any domain (DC, core, metro, access network), while hiding the details from the applications utilizing NIDO. So, NIDO is more robust and flexible in comparison to network/technology-specific related systems that have been presented [4].

At its North Bound Interface (NBI), NIDO exposes a REST APIs to enable the integration with cloud orchestrators (e.g. OpenStack), allowing to setup and tear-down quality of service (QoS) guaranteed end-to-end connections as part of the whole workflow for cloud service provisioning, termination, migration or recovery. For example, Figure 2 shows the workflow for the provisioning of a dedicated inter-domain transport network connection between the servers where the VMs are placed. Initially, when a new domain is added through NIDO there is a process for building the multi-domain network topology (Step 1-2). The network provisioning is initiated by a cloud orchestrator that triggers the entire automated procedure (Step 3-6) to establish an end-to-end path from a server in DC-1 to a server in DC-2, through which the VMs traffic will be carried. This approach allows to hide the internal details of the network topology, protocols and configuration mechanisms from the cloud controller, which uses only abstract and technology-agnostic, domain-independent interfaces (usually REST-APIs) to request connections with the desired QoS parameters.

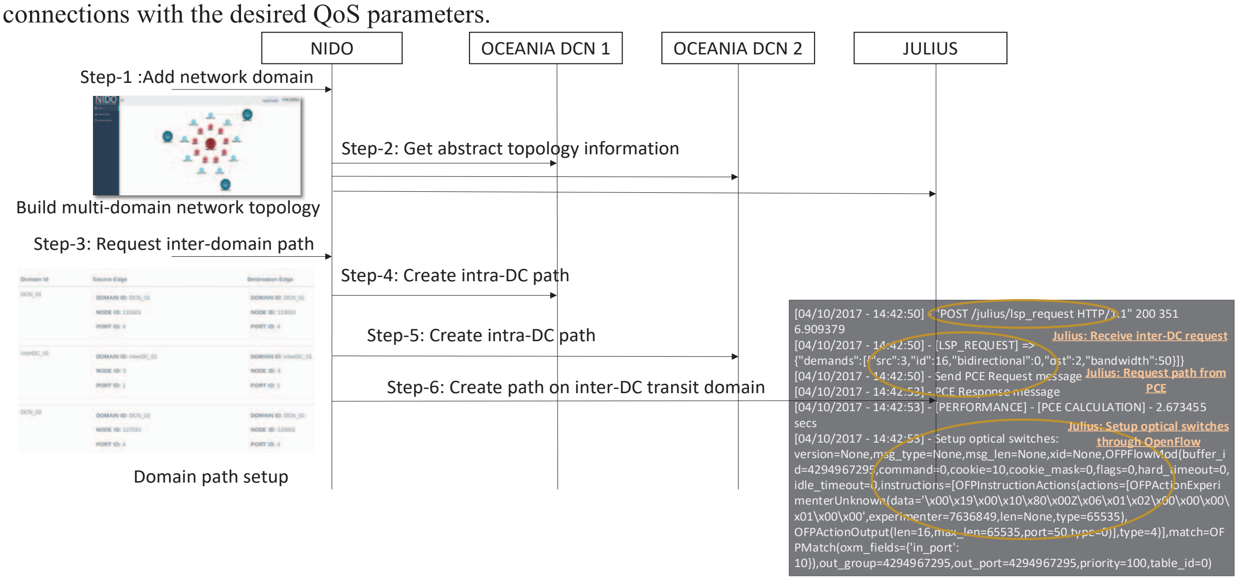


Figure 2: Workflow for provisioning of a dedicated inter-domain transport network through NIDO.

NIDO's internal procedure for provisioning an end-to-end network service consists of two main steps. NIDO computes initially the domain path for the requested inter-domain connection, i.e. the sequence of domains that must be traversed, together with the edge nodes at each domain. In the second step NIDO issues requests to setup the intra-domain network connection in the domains returned in the domain path, utilizing the NBI of the respective intra-domain controllers. (i.e. OCEANIA for DCN and JULIUS for the inter-DC transport network). Once all the intra-domain paths have been established and acknowledged to NIDO, the inter-domain path becomes active. NIDO is responsible for maintaining the status of the different inter-domain paths, together with their relationship to the list of intra-domain paths that compose them. The network topology is also kept updated accordingly, modifying the resource availability on the inter-domain links based on the active inter-domain paths. Also, through NIDO it is possible to add and remove domains dynamically, specifying the type of plugin that must be used to interact with the associated controller. NIDO software prototype has been developed in Java and released as open source software[1].

OCEANIA intra-domain orchestrator for NEPHELE-based DCN [6] is based on the OpenDaylight SDN controller. OCEANIA has the complete and detailed view of the NEPHELE optical time-slotted DCN, composed of hybrid ToR and POD switches and hosts. OCEANIA implements technology-specific mechanisms for setting up and tearing down network connections within the controlled DC, using algorithms for the global optimization of the intra-DC resource allocation. The resource allocation decisions are then translated into a set of OpenFlow rules that

[1] https://github.com/nextworks-it/nephele-nido

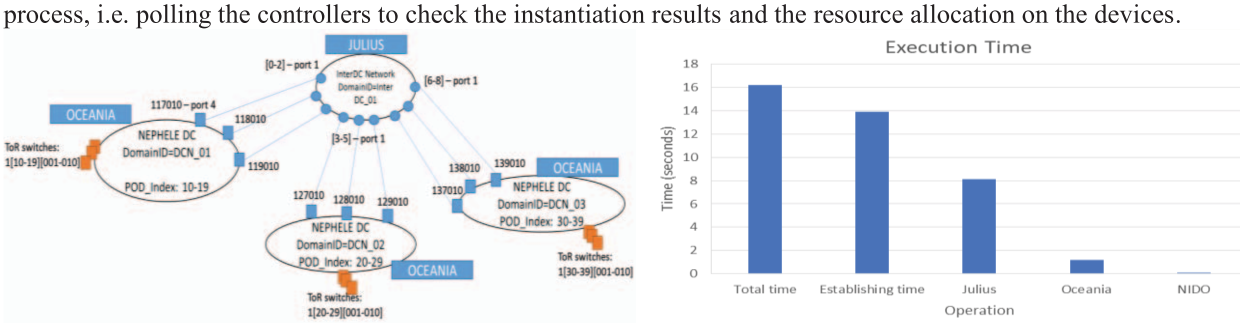
are then installed into the DCN devices (ToR and POD switches), configuring the forwarding behavior at the data plane level. OCEANIA is also released as open source software[2] and has been publicly demonstrated at [6].

The inter-DC network domain is managed by the JULIUS orchestrator [7], which implements mechanisms to establish connections between respective edge nodes, extending Ryu SDN-controller for interacting with network devices (both IP and Optical) via appropriate protocols (OpenFlow and OpenFlow with Optical Extensions). Actually, Julius is both an inter-DC controller and a complete emulation platform for SDN enabled, multi-layer IP/Elastic Optical Networks. Emulation is enabled using Mininet, Open Virtual Switches to emulate IP/L3 switches and LINC-OE to emulate fixed and flexible optical nodes. Julius also integrates Mantis' stateful Path Computation Element (PCE), providing optimization logic for path and spectrum allocation in multi-layer IP/Optical networks.

3. Testbed and Results

NIDO, along with OCEANIA and JULIUS, were deployed and functionally validated in a test-bed using OpenStack as cloud platform, with a Mininet-based emulated, inter-domain network (Figure 3.a), with 3 intra-DC and 1 inter-DC network domains of variable sizes. This emulated approach allowed to easily scale the dimension of the network topology, in order to verify the behavior and the scalability of the system. The internal topology of each DCN includes 3 optical planes, each of them with 10 POD switches. Each POD switch is in turn connected to 10 ToR switches. In each DC, the three ToR switches act as gateways towards the inter-DC network. The inter-DC network includes 9 core nodes, while 3 edge nodes connect to each DCN.

The experiments were based on multiple inter-domain requests to setup and tear-down end-to-end connections between hosts located in different DCs, triggered through the NIDO NBI. The network orchestrator decomposes the requests in the inter-DC network request part (source and destination DCs) and inter-DC networks requests, triggering requests to the NBI of each respective controller. In Figure 3.b we report the time required for NIDO to perform each sub-task of the orchestration process. We can see that the computation time is negligible, as NIDO operates on an abstract view of the network which consists only of DCNs/inter-DC networks as nodes and border links as edges. Most of the path instantiation time (time between the first request sent to a domain controller and the last intra-domain path activation) is spent establishing the paths in the single domains. The actual time between the request's arrival and the path activation is about 16s which includes also the overhead intrinsic in the orchestration process, i.e. polling the controllers to check the instantiation results and the resource allocation on the devices.



(a) (b)

Figure 3: a) Virtual network topology for inter-DC testbed, b) Execution time for resource allocation.

4. Conclusions

We present a hierarchical orchestration platform for inter-domain DC networks that includes hybrid electrical/optical intra-DC networks and an elastic optical inter-DC network. The platform was functionally validated by demonstrating dynamic and end-to-end allocation of capacity in an emulated inter-domain testbed.

Acknowledgment

This work was partially supported by the EC through the Horizon 2020 Nephele project (g.a. 645212).

5. References

[1] "Cisco Global Cloud Index: Forecast and Methodology, 2015–2020", Retrieved October 2017.
[2] P. Kokkinos, et al. "Survey: Live Migration and Disaster Recovery over Long-Distance Networks", ACM Computing Surveys, 2016.
[3] K. Christodoulopoulos, et al., "Elastic bandwidth allocation in flexible OFDMbased optical networks," JLT, 2011.
[4] H. Yang, et al., "SUDOI: software defined networking for ubiquitous data center optical interconnection," IEEE Communications Magazine, vol. 54, no. 2, pp. 86-95, 2016.
[5] NEPHELE EU H2020 project, www.nepheleproject.eu
[6] G. Landi, et al., "SDN Control Framework with Dynamic Resource Assignment for Slotted Optical Datacenter Networks," OFC 2017.
[7] A. Kretsis, et al., "An emulation environment for SDN enabled flexible IP/optical networks," ICTON 2016.

[2] https://github.com/nextworks-it/oceania-dcn-controller