Broker-assisted Multi-domain Network Service Orchestration*

Danny Alex Lachos Perez[1] and Christian Esteve Rothenberg[1] and Róbert Szabó[2]

***Abstract*—Future 5G scenarios call for service orchestration models beyond the achievements of ETSI MANO. Complex multi-vendor, heterogeneous technology and resource environments require elaborated collaboration mechanisms between different network operators. In this paper, we present a broker-plane approach on top of per-domain management and orchestration functions to coordinate the delivery of a multi-operator End-to-End Network Service (E2ENS) that combines per-domain paths and network functions for a given service request. In our prototype implementation, a broker retrieves local information about topology and resources from each Multi-domain Orchestrator (MdO) to offer after added value services in the form of abstract Map Services. These map abstractions are easily consumable through developer-friendly REST APIs based on the Application-Layer Traffic Optimization (ALTO) standard protocol.**

***Index Terms*—5G, Orchestration, NFV, ALTO**

I. INTRODUCTION

Envisioned 5G network architectures and related service models consider wider cooperation between stakeholders in order to provide flexible multi-operator multi-domain services [1]. Deeper collaboration between different network operators is critical to enable new business model approaches, including federation models [2]. In this work, we propose a federation networking paradigm where a broker-plane on top of Multi-domain Orchestrators (MdOs) assists the coordinated creation of an End-to-End Network Service (E2ENS). In our proposed brokered architecture, each MdO involved in the federation advertises to the federation layer (acting as a broker) the intra-domain resource and topology information. From this local information, the broker creates an aggregated inter-domain information exposed as a set of abstract and unified Map Services accessible to the MdOs through ALTO-based REST APIs [3].

Benefits that follow the proposed brokering layer include:

- Scalability. Avoid the distribution of topology and resource information in a peer-to-peer fashion (MdO-to-MdO). The more MdO-to-MdO interconnections, the larger the "costs" of distribution.
- Flexibility. Allow domains without physical infrastructure (e.g., without BGP or BGP-LS instances) to advertise and learn (avoiding the deployment and configuration of per domain BGP peering points).
- Complexity. Leveraging the joint inter-domain topology information to pre-select a (sorted) list of candidate domains reduces the number of redundant links along the path(s) in E2ENS requests.

*This work was supported by the Innovation Center of Ericsson, Brazil.

[1]D. A. L. Perez and C. E. Rothenberg are with the School of Electrical and Computer Engineering (FEEC), University of Campinas (UNICAMP), São Paulo, Brazil dlachosp, chesteve at dca.fee.unicamp.br [2] Róbert Szabó is with Ericsson Research, Hungary. robert.szabo at ericsson.com

II. ARCHITECTURE

As shown in Figure 1, a set of Domain Orchestrators (DOs) responsible for various resource domains (featuring physical and virtual, software and hardware components) is assumed to be managed by a MdO. The broker layer is conceived to be working as a coordinator between a set of MdOs. The main architectural components are described next:

1) Inter-domain Resource (IdR): It creates a hierarchical database that contains inter-domain resource information such as resource availability (i.e., CPU, memory, and storage), virtual/physical Network Functions (NFs) supported and Service Access Points (SAPs) to access those resource. UNIFY [4], TOSCA [5], among other data models can be used to define the interface between IdR and MdOs.

2) Inter-domain Topology (IdT): A hierarchical TED (Traffic Engineering Database) that contains inter-domain network topology information, including additional key parameters (e.g., MdO entry point). This information can be retrieved from each MdO through BGP-LS or REST interfaces.

3) Map Service: This component is the core of the Broker layer. The information collected from the IdR and IdT modules are processed here to provide abstract maps with a simplified view, yet enough information about MdOs involved in the federation. The goals of the Map Service is to convey: (*i*) a set of properties (e.g. SAPs, NFs, IT resources, etc.) for MdOs in the federation (*Property Map*), (*ii*) domain-level topology (*Topology Map*), and (*iii*) path costs for an E2ENS request (*Cost Map*). All these maps are provided through a common transport and encoding protocol as defined in ALTO based on existing HTTP technologies (RESTful and JSON).

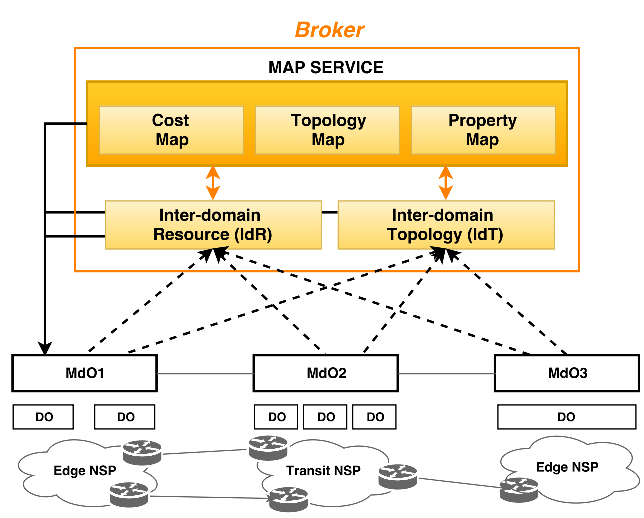

Fig. 1: Broker-assisted Multi-operator Network Architecture

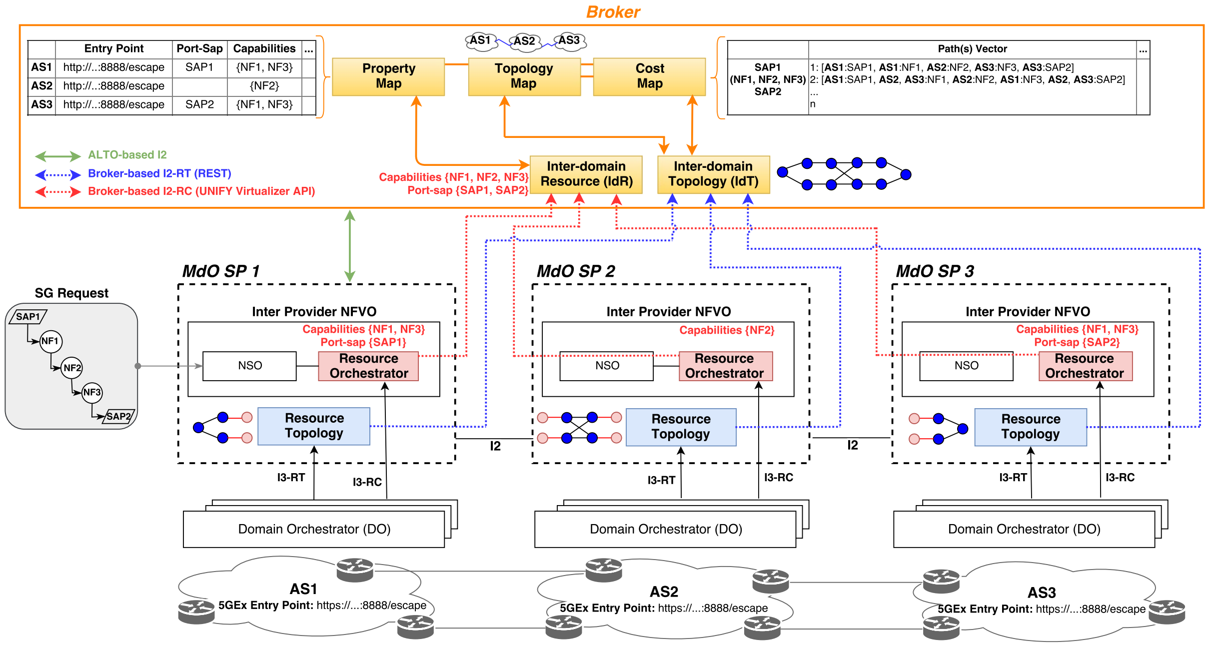

Fig. 2: Broker-assisted 5GEx Information Exchange

III. Proof of Concept Use Case Implementation

The strawman use case scenario refers to an E2ENS orchestration involving 3 Service Providers (SPs), as shown in Fig. 2. Each SP has a MdO[1] to coordinate resource and/or service orchestration at multi-operator level via interface I2 APIs. For the orchestration within the same administrative domain, each MdO uses emulated DOs with emulated I3 interfaces, since no data-plane is present. DOs use static configuration files to load local information about resources (I3-RC) and topology (I3-RT). The different MdO components are based on existing open source tools such as ESCAPE[2] (Resource Orchestrator) and Netphony-topology[3] (Resource Topology) and run in Docker containers on a single computer.

In case of the broker layer, the IdR and IdT components use the UNIFY Virtualizer API [4] (broker-based I2-RC API) and REST API (broker-based I2-RT API) respectively, in order to create the hierarchical databases. From this inter-domain information, the three main Map Services are created:

- The *Property Map* includes property values grouped by Autonomous System (AS). Such values include SAPs, NFs and the 5GEx Entry Point (e.g., the URL of the ESCAPE).
- The *Topology Map* module creates an AS-level connectivity graph from the information in the IdT component.
- The *Cost Map* defines a path vector as an array of ASes, representing the AS-level topological distance between entities (i.e., AS→AS, SAP→SAP, NF→NF or SAP↔NF). Path vector constraints can be applied to restrict the response to costs that satisfy a list of simple predicates (e.g., $=, >, <, \geq, \leq$). Even, a special *"shortest"* predicate can be used to provide the shortest path between entities.

The Map Service component has been derived from the OpenDayLight ALTO[4] framework and the Neo4j[5] graph-based database is used as the back-end.

Finally, when a MdO receives a Service Graph (SG) request, it uses the broker component (through ALTO-based I2 APIs) to be able to determine the underlying network graph and a potential set of paths before bilateral negotiation between MdOs are started.

IV. Conclusions and Future Work

This work proposes a broker-assisted approach with a set of potential benefits to the challenges of multi-domain orchestration by leveraging the map abstractions and generality of the ALTO protocol. In addition to open sourcing, our future activities include to pursue contributions to standardization along (i) the way MdOs describe their network capabilities, and (ii) extensions to the base ALTO protocol as necessary.

References

[1] H2020 Vital Project, "D4.3 Federated Network Resource Manager (FNRM) Design," 08 2016. [Online]. Available: https://goo.gl/9r5xwh

[2] 5G-PPP, "Advanced 5G Network Infrastructure for the Future Internet," 11 2013. [Online]. Available: https://goo.gl/CJHVNf

[3] R. Alimi *et al.*, "Application-layer traffic optimization (alto) protocol," Internet Requests for Comments, RFC Editor, RFC 7285, September 2014. [Online]. Available: http://www.rfc-editor.org/rfc/rfc7285.txt

[4] UNIFY Deliverable D3.2a. (2015) NFFG specification. [Online]. Available: https://goo.gl/8RA7HN

[5] OASIS. (2013) TOSCA specification. [Online]. Available: http://docs.oasis-open.org/tosca/TOSCA/v1.0/os/TOSCA-v1.0-os.pdf

[1]The functional components and interfaces of the MdOs follow the architectural proposal of the 5GEx project. http://www.5gex.eu/wp/?page_id=510

[2]https://github.com/5GExchange/escape

[3]https://github.com/telefonicaid/netphony-topology

[4]https://github.com/opendaylight/alto

[5]https://neo4j.com/