Mobile Transport and Computing Platform for 5G Verticals: resource abstraction and implementation

Mobile Transport and Computing Platform for 5G Verticals: Resource Abstraction and Implementation

N. Sambo, L. Valcarenghi
Scuola Superiore Sant'Anna, Pisa, Italy

A. Garcia-Saavedra
NEC Laboratories Europe, Germany

I. Pascual, R. Martinez, J. Mangues-Bafalluy
Centre Tecnològic de Telecomunicacions de Catalunya
CTTC/CERCA, Spain

A. Ksentini
EURECOM, France

P. Iovanna, G. Imbarlina, F. Ubaldi, T. Pepe
Ericsson, Pisa, Italy

G. Landi
Nextworks, Italy

C. Vitale, C. Chiasserini
Politecnico di Torino, Italy

F. Klamm
Orange, France

C. Turyagyenda
InterDigital, London

***Abstract*—The 5G-TRANSFORMER project aims at transforming today's mobile transport networks into an SDN/NFV-based platform, which offers slices tailored to the needs of vertical industries. The paper describes the 5G-TRANSFORMER resource management layer, namely MTP, and its main functionalities such as resource abstraction, resource information modeling and orchestration, and service instantiation. Then, it focuses on the ETSI-based interfaces exploited for the interaction between the control and management plane elements. Finally, the paper reports an MTP implementation including messages reporting resource abstraction.**

Keywords—mobile, transport, computing, abstraction, interfaces

I. INTRODUCTION

The 5GPPP/H2020 5G-TRANSFORMER project [1] defines control building blocks to offer mobile transport network and computing slices tailored to the needs of vertical industries: the Service Orchestrator (SO) and the Mobile Transport and Computing Platform (MTP). The SO offers end-to-end service orchestration and federation of transport networking and computing resources from multiple domains. The MTP coordinates the underlying unified mobile, transport, and computing stratum on which vertical services are deployed. Based on proper information modeling, MTP provides resource abstraction and orchestration, and service instantiation.

According to 5G paradigm, processing and storage resources dedicated for vertical applications could be distributed in datacenters that are connected to each other by the transport and mobile network resources. Hence, the MTP has been defined as a novel block that manages the complexity of multi-domain transport, mobile, and data center resources providing a suitable abstract view to the SO. This approach allows to decouple the operations on the networking and data center resources (that are delegated to the MTP) with respect to the virtual function placement for vertical applications (delegated to the SO). MTP acts as a single point towards the SO providing the abstract view of the mobile, transport, compute, and storage resources and, at in turn, translating the requests from the SO in resource requests to the transport, mobile, and data centers segments.

This paper is focused on the MTP, which leverages on the ETSI NFV [2][3], a standard used as reference architecture adopted by the 5G-TRANSFORMER project [1], here extended to deal with the complexity of the transport and mobile networking, as well as compute and storage resources. Actually, the MTP fills some gaps not covered by the current interfaces defined by ETSI, such as resource abstraction and orchestration of different technological and administrative domains.

Finally, the paper reports an implementation of the MTP, showing the structure of the control and management messages for topology abstraction and service request.

II. RELATION WITH ETSI

The MTP can be seen as an enhanced Virtualised Infrastructure Manager (VIM), here abstracting and orchestrating different technological and administrative domains. As reported in the ETSI-MANO document [4] the "…Virtualised Infrastructure Manager (VIM) is responsible for controlling and managing the Network Function Virtualization Infrastructure (NFVI) compute, storage and network resources, usually within one operator's Infrastructure Domain (e.g. all resources within an NFVI-PoP, resources across multiple

NFVI-POPs, or a subset of resources within an NFVI-PoP)...". Thus, VIM orchestrates the allocation and release of NFVI resources, manages the repository of both hardware (compute, storage, and network) and software (e.g., hypervisors) resources, and collects performance and fault information.

The interfaces of ETSI IFA005 and IFA006 are here adopted for abstracting and orchestrating compute, storage, network and mobile resources within each domain. ETSI IFA005 [5] and IFA006 [6] define standard interfaces to enable a consumer block at the VIM north bound interface to retrieve information about resources capability (e.g., maximum CPU), to allocate and reserve a resource, and to monitor a given performance metric with the objective of maintaining a vertical service. Each interface is associated to one of three types: *Resource Database* (an interface exploited to retrieve information about the capabilities and resource availability in the NFVI), *Reservation and Management* (exploited to reserve, allocate, or release a resource), and *Maintenance* (exploited for performance monitoring and service maintenance).

III. Mobile Transport and Computing Platform (MTP)

This section describes the control and management architecture (shown in Figure 1) and highlights the main novelties introduced by 5G-TRANSFORMER into the MTP.

A. Overvall architecture

The architectural design of the MTP aims at providing a set of functionalities and operations to support the Service Orchestrator (SO) in the efficient utilization of different NFVI infrastructure domains, following a NFVI-as-a-Service model.

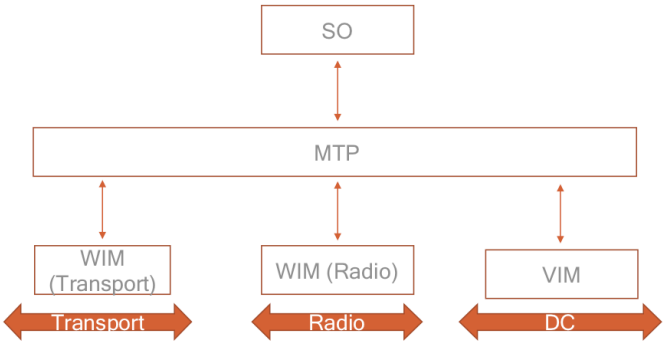


Figure 1 Control and management architecture

MTP orchestrates resources selected for the resource allocation request of the SO. MTP functions include the instantiation of virtual network functions and the management of the underlying physical mobile and transport network, computing and storage infrastructures. The latter may be deployed in central data centers as well as distributed. The MTP provides support for slicing and enforces slice requirements coming from the SO as well as physical infrastructure monitoring and analytics services. The abstraction functionalities of the MTP allow for different levels of resources abstraction to be offered to the SO. Then, MTP coordinates VIM (for the control and management of data centers – DC – thus, for compute, storage, and connectivity within a DC), and the Wide Area Network Infrastructure Managers (WIMs). The WIM is a control module standardized by ETSI [4] responsible for the control and management of connectivity in the wide area network. Here, we assume WIM to control mobile and transport networks.

B. MTP main novelties

The MTP, as the overall 5G-TRANSFORMER architecture, has been designed to be aligned with the ETSI NFV specifications. Anyway, extension of ETSI specifications is done to support the goals of the project.

One innovation lies on the fact that service and resource orchestration of the ETSI architecture are partitioned among SO and MTP. SO works on an abstract view provided by the MTP, where the complexity of the transport and radio mobile networks is lightened by exposing logical links connecting the data-centers resources dedicated for vertical applications. The MTP manages all the complexity of the transport, mobile, storage and compute resources, providing, besides a suitable abstraction, also the configuration of such resources. Moreover, internally, the MTP decouples the transport, mobile and data center resources to assure that each of them could be owned and managed by different business actors. Such decoupling also facilitates the development of an MTP architecture where a single MTP can integrate several VIMs and WIMs from different technological domains and exposes a unified view to the upper layers (e.g., to the SO). The integration of several VIMs and WIMs allows a single entity to control several technological domains.

In order to allow the integration of several VIMs and WIMs in one MTP, the MTP includes an abstraction layer, which in turn is able to provide different levels of abstraction at both cloud computing and networking levels. Depending on the level of details exposed to the upper layer, the MTP may take autonomous decisions about resource orchestration or these decisions can be taken by the SO.

The integration of multi-access edge computing in the 5G-TRANSFORMER project has also its reflection in the MTP, since it must be able to support: (i) advertisement of the hosts, including their characteristics (locations, capabilities, network connectivity to radio access network – RAN – and WIMs); (ii) deployment of applications and configuration of the related traffic steering; (iii) advertisement of services running in each host; (iv) support of network interfaces towards the RAN and the data plane in general.

The MTP can also deploy, manage and provide a mobile service, including the combination of network functions and connectivity from the RAN to the transport. This kind of service is offered as a resource to the upper layer (i.e., the SO) and it is managed autonomously by the MTP itself. This means that the MTP is able to select, deploy and configure the most suitable RAN split, as well as to decide the internal decomposition of such service, e.g. using physical or virtual network functions, and its dimensioning.

IV. MTP INTERFACES AND ABSTRACTION

The MTP uses two sets of interfaces: an external Northbound interface (NBI) between MTP and SO and a Southbound Interface (SBI) between VIMs and WIMs. These two interfaces are described in the following subsections *A* and *B*, while subsections *C* and *D* present abstraction and some use cases, respectively.

One of the main innovations of the MTP is related to the different levels of abstraction that it exposes to the SO through the NBI for the control and advertisement of the NFVI resources. This, in turn, implies the definition of information models.

A. NBI

The MTP NBI addresses the interworking between the SO and the MTP. To be aligned with [5], from a high level view, the MTP NBI must provide three main functionalities: i) virtualized resource information management, ii) virtualized resource management and iii) virtualized resource fault and performance management.

The *virtualized resource information management* allows the SO to retrieve information about the available, allocated, and reserved virtualized resources encompassing compute, storage and networking. Such information can be delivered by using different levels of abstraction. Thus, the adopted abstraction will notably impact the SO algorithms used for the virtual network function placement and/or networking computation. The mechanism used to update virtualized resource information toward the SO could be achieved via different mechanisms such as immediate update when a change in any (abstracted) resource occurs (e.g., allocation or reservation), upon explicitly request sent by the SO, or even applying predefined periodic updates.

The *virtualized resource management* focuses on enabling the SO to perform the operations for allocating, reserving, updating, and releasing virtualized resources. The explicit selection of the virtualized resources, for instance when a new network service is being created, strongly depends on the level of abstraction of the resources exposed by the MTP. In other words, if the virtualized resource information at the SO is highly abstracted, the final selection of the resources should be delegated to the underlying MTP functional block. In this regard, the MTP NBI interface must allow the SO to define the request for specific resource allocation constraints that the MTP selection should consider, such as location constraints, resource groups associated to a particular partition or tenant, affinity or anti-affinity constraints, etc. The ultimate goal is that, regardless of SO resource visibility, this functionality must end up with the actual management of the resources handled by MTP and consumed by the network services.

The *virtualized resource fault and performance management* functionality is meant to handle alarms when an error / fault occurs specifying the set of impacted virtualized resources. To do that, the SO should have an (abstracted) view of the potential resources being impacted by an error/fault, to trigger a candidate recovery action if the MTP is unable to recover the fault by itself.

Besides handling fault events, the MTP NBI should also allow the SO to collect performance information about the virtual resources. This information is related to measurements describing the consumption level of allocated virtualized resources such as the CPU power consumption, the disk latency, and the average bandwidth used in a link. This information allows the SO adopting strategies and actions to anticipate potential service level agreement violations or the need to scale resources for a virtual network function. To retrieve these measurements collected at the MTP context, the SO should make a subscription for the set of performance measurements it is interested on, specifying the target virtualized resources along with other details about the periodicity when the MTP should collect the information and trigger the notifications towards the SO.

B. SBI

Because the MTP embeds not only the VIM functionalities but also NFVI functionalities, the MTP southbound interface is an internal interface that interacts with the different VIM/WIM technological domains. In particular, the MTP interacts with the VIM for storage and compute resources, while with the WIM to achieve connectivity in the wide area network. MTP exploits the functionalities to interact directly with the infrastructure network domain, the hypervisor domain, and the compute domain of an NFVI. Moreover, MTP is able to retrieve the infrastructure resources to compute internally the abstraction to export using the NBI interface.

C. Abstraction levels and information model

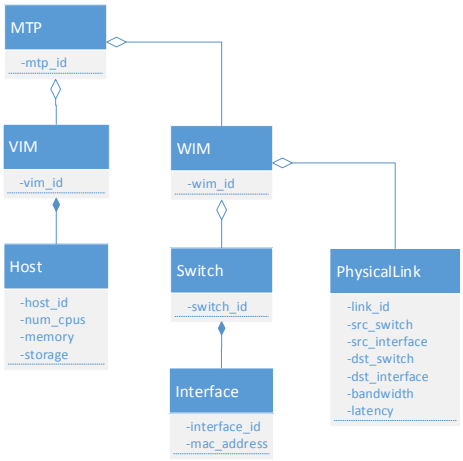
The MTP is responsible for providing the SO with the information about the available resources, so that the SO can make decisions on service instantiation. Because of the varying level of trust among organizations and the complexity associated to resource management, the MTP, in general, does not provide all of its infrastructure details. Rather, it presents to the SO the information with a certain *level of abstraction*.

Specifically, the resources managed by a MTP can be divided in two groups: computing resources and network resources. Computing resources are the physical machines that can accommodate virtual network functions (VNFs) and are typically characterized by CPU, memory, and storage capabilities. Computing resources are grouped depending on the location in NFVI Points of Presence (NFVI-PoPs). The physical machines of an NFVI-PoP are managed by the VIM, actually managing computing resources. Network resources are represented by the network forwarding units and the physical links interconnecting them. WIMs control network resources connecting different NFVI-PoPs.

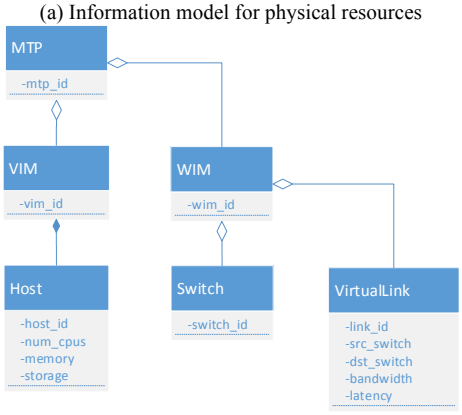
An infrastructure can thus be represented as a composition of network and computing (including storage) resources controlled by WIMs and VIMs, respectively. Since the nature of these resources is intrinsically different, the abstraction mechanisms for these two types of resources can also be different and can be combined as follows. Figure 2 shows basic information models for the presented abstraction levels plus an information model for the physical infrastructure

(Figure 2a). These information models can be extended, e.g. to include location information or cost.

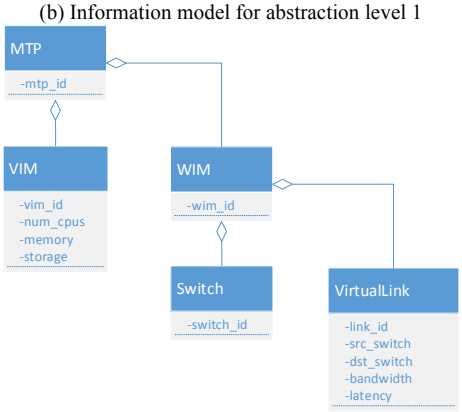
Level 1: also named Network level because only network resources are abstracted (Figure 2b). The MTP reports all details about computing resources while the network resources are abstracted as a set of logical links connecting the physical machines, with each link being characterized by a given bandwidth and latency.



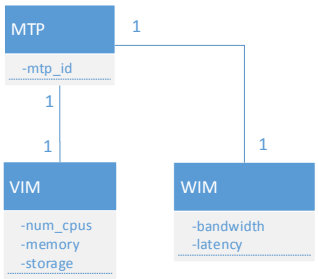
(a) Information model for physical resources



(b) Information model for abstraction level 1



(c) Information model for abstraction level 2



(d) Information model for abstraction level 3

Figure 2 Information models

Level 2: also named Compute level (Figure 2c) because, besides the network abstraction of level 1, the computing resources are aggregated per NFVI-PoP. The MTP reports the computing capabilities, CPUs, memory, storage, with an NFVI-PoP granularity instead of by physical-machine granularity as in Level 1. Regarding the network resources, only the connections between NFVI-PoPs are reported, as virtual links with a given bandwidth and latency.

Level 3: also named MTP level (Figure 2d) because all resources, both computing and network resources, are aggregated with MTP granularity. This level may be useful for resource federation, as it allows a SO to expose to peer SOs the resources available within its administrative domain while hiding the complexity and the infrastructure details. In general, this higher level of abstraction is handled by the SO, as it is the one to decide which levels of abstraction to be exposed to other SOs, due to administrative or agreement on information constraints.

The information model for abstraction level 1 (Figure 2b) shows how the physical links have been converted to virtual links and thus the interface information of the switches is no longer needed. The information model for abstraction level 2 (Figure 2c) shows how the hosts representation is no longer used and the compute, memory and storage resources are aggregated at a NFVI-PoP level. Finally, the abstraction level 3 (Figure 2d) represents the MTP as one single NFVI-PoP and one single network where all NFVI-PoPs and networks resources are aggregated respectively.

D. Vertical: use cases of application

This section presents two use cases of MTP abstraction for verticals: i) automotive; ii) cloud robotics. Regarding automotive, connected cars enable a multitude of new applications, among which, applications involving road safety. An example of these applications is collision avoidance at urban intersections, which is referred to as ICA hereinafter. Over a specific monitored area, connected cars may indeed transmit information related to their position, speed and direction and, with this type of information, a collision avoidance application may take prompt actions if a dangerous situation is predicted [7]. In 5G-TRANFORMER, we move the intelligence of the collision avoidance application from the car to the infrastructure and we consider Vehicle-to-Infrastructure (V2I) communications. Importantly, the

infrastructure supporting the ICA application has to comply with several requirements, among which: (i) reliable coverage over the monitored area, to enable correct delivery of alarm messages; (ii) highly reliable positioning accuracy; (iii) strict latency, in order to promptly take actions when a dangerous situation is detected. No matter which level of abstraction is used by the MTP, the SO chooses the location of the different Virtual Applications (VAs) of the ICA so that such requirements are satisfied. For example, due to the aforementioned latency constraint, the SO most likely will exploits NFVI-PoPs which are very close to the monitored area, making the ICA application a strong candidate for a Multi-access Edge Computing (MEC)-based implementation.

As far as the cloud robotics use case is concerned, the industrial automation can be considered as a possible application scenario. Highly automation of the factory plant is provided moving the control of the production processes and of the robots' functionalities in cloud, exploiting wireless connectivity to minimize infrastructure, optimize processes, implement lean manufacturing. As for the ICA application, moving the intelligence to the infrastructure, it is important to satisfy several requirements, in particular: (i) low latency to allow a correct operation of the robotic area; (ii) reliability to guarantee successful message transmissions within a defined latency budget or delay; (iii) "5-nines" availability on wireless links. Consequently, the applications (i.e., Vertical Applications --- VAs) composing the cloud robotics service must be conveniently deployed by the SO to satisfy the service requirements. For instance, non-time-critical services, like the system in charge to manage and control the plant, can be located remotely. Time-critical services, instead, like navigation or data processing have been kept close to the radio base station (preferring a MEC-based implementation) to guarantee low latency, stability and a proper reaction time.

As shown in Section IV, the MTP may represent the actual available resources with different levels of abstraction. Here, we illustrate the applications example with abstraction level 1 and level 2. When the MTP uses level 1 abstraction, it presents the resources as follows:

- the network resources, i.e., all the (virtual) links: such information is retrieved thanks to the interaction with the WIMs. The MTP hides the complexity of the network connectivity showing only the logical connectivity between the access network and the NFVI-PoPs, or between NFVI-PoPs. Each (virtual) link is characterized with its latency and bandwidth. Mean and max performance statistics of each (virtual) link, also the ones involving the radio access network, are presented during the abstraction.
- the computation and the storage resources, both at the MEC and at the Cloud NFVI-PoPs: such information are retrieved thanks to the interaction with the VIMs. In the abstraction level 1, the full connectivity between processing units within the NFVI-PoPs is also presented.

Over this abstracted view of the resources presented by the MTP, the SO places the different VAs of the specific application e.g., the SO instantiates the Collision Avoidance Server (CAS) where the algorithm for the collision avoidance runs as an application server for the ICA application or select the server where running the control functionalities for the CR application. While the networking latency is directly exposed by the MTP, the SO assesses the processing delay of the VAs, which not only depends on the physical characteristics of the processing units, but also from the computational demands of the VAs tasks.

Finally, Figure 3 shows a possible abstraction representation of available resources at the MTP with abstraction level 2. Here, the resources of the NFVI-PoPs are presented as the aggregation of different processing units. The key difference with abstraction level 1 is that the MTP is responsible of choosing, within the NFVI-PoP, the processing unit where to allocate the assigned VAs. The choice of the MTP is done respecting the QoS of the specific application.

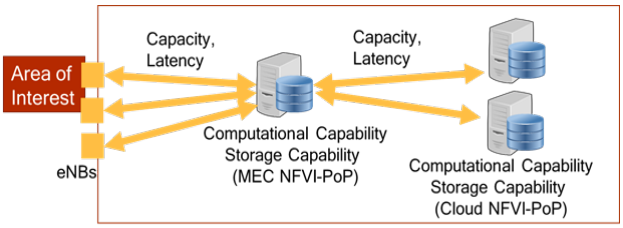


Figure 3 A possible example of abstraction for the ICA application

V. MTP IMPLEMENTATION

In this section, we present a first implementation of the MTP, which runs as a standalone application opening multiple sockets towards other 5G-TRANSFORMER functional elements: a Northbound socket (NBI_Sock) towards a SO and multiple Soutbound sockets towards WIM (SBI_Sock). The SO runs on a PC opening a NBI_sock toward the MTP (which is on a different PC). The allocation and termination of resources managed by the WIM is reported. Specifically, MTP knows the WIM resources and computes an abstraction of them, which finally is exported to the SO. The abstraction represents the resources including information about service parameters (like bandwidth, delay, jitter). The internal complexity of the resources (like physical and technological specific parameters, control specific details, and so on) is hidden to SO. The demo is triggered by SO requesting to MTP the allocation/termination of a service using the abstracted parameter; according to that, MTP allocates and terminates the resources, and, finally, notifies SO about the outcome.

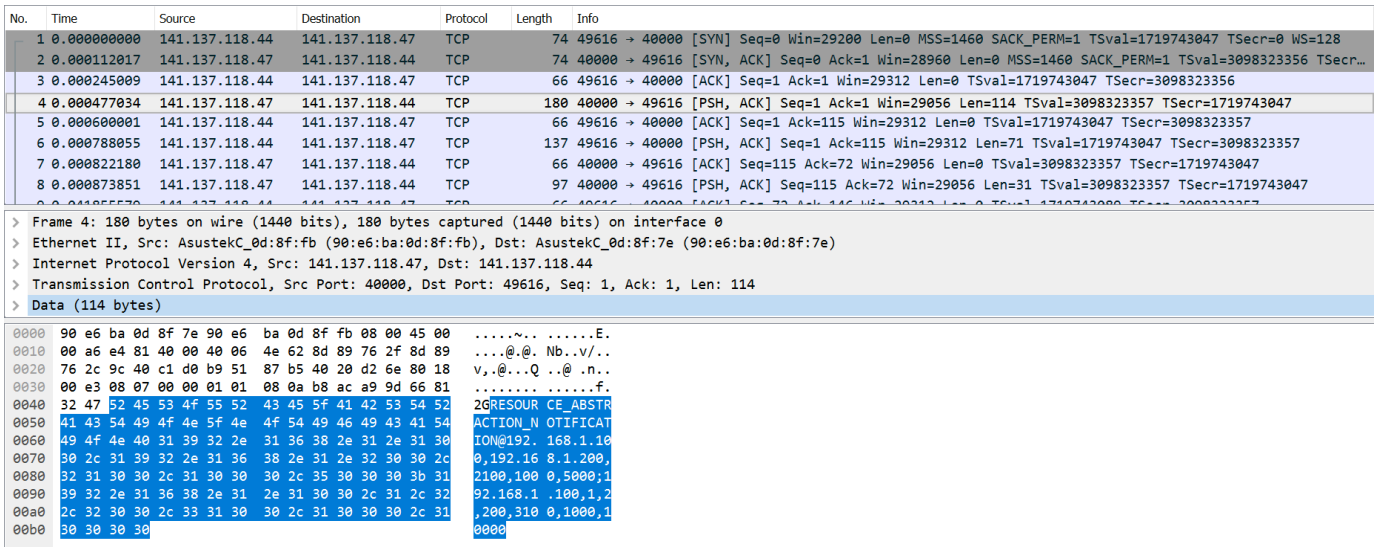


Figure 4 Wireshark capture of control and management messages

The exchanged custom-built messages are Remote Procedure Call based and are sent on a TCP client server communication (SO is the client while the MTP is the server). A Wireshark capture of the control and management messages is shown in Figure 4, while their format is presented in Figure 5. In particular, Figure 5a shows the message exchanged by MTP to notify the topology abstraction to SO (e.g., the bandwidth in Mbit/s and the delay in nanoseconds are included). Figure 5b represents the service request including source-destination pair identifiers and the requested rate with information about tolerated delay. Finally, Figure 5c reports the result of the service allocation (outcome).

(a) Notification from the server to the client: topology abstraction:

RESOURCE_ABSTRACTION_NOTIFICATION@LogicalSrcRouterID, LogicaldstRouterID, LogicalLinkId1, BW1(Mbit/s),Delay1(us);…..; srcIPn,dstIPn,LogicalLinkIdn, BWn(Mbit/s), Delayn(us)

(b) Service request from the client to the server:

RESOURCE_ALLOCATION_REQUEST@ServiceID, LogicalSrcRouterID, LogicaldstRouterID, LogicalLinkId, BW(Mbit/s), Delay(us)

(c) Reply to service request:

RESOURCE_ALLOCATION_REPLY@ServiceID, Outcome

Figure 5 Control and management messages

Figure 4 also shows (highlighted at the bottom) the payload of message 4, which represents the request of topology abstraction, is highlighted at the bottom of Figure 4. Just to clarify, in this example SO and MTP are in the same sub-network: SO runs on a PC with IP 141.137.118.44 and MTP on a different machine with IP 141.137.118.47.

VI. CONCLUSIONS

This paper presented the Mobile Transport and Computing Platform (MTP) of the 5G-TRANSFORMER architecture for vertical application support in 5G network scenarios. The MTP performs resource orchestration and abstraction to support a service orchestration guaranteeing verticals' requirements. MTP interacts with VIMs and WIMs triggering the management of computing and network resources, respectively. An implementation of MTP is presented in this paper showing the control and management messages reporting resource abstraction, service request, and the notification of the allocation of a service.

ACKNOWLEDGMENT

This work has been partially funded by the EU H2020 5G-TRANSFORMER Project (grant no. 761536).

REFERENCES

[1] www.5G-transformer.eu

[2] ETSI GS NFV 002 v1.2.1, "Network Functions Virtualisation (NFV); Architectural Framework", Dec. 2014

[3] ETSI GS NFV-INF 001 V1.1.1 (2015-01), "Network Functions Virtualisation (NFV); Infrastructure Overview". Jan. 2015

[4] ETSI GS NFV-MAN 001 V1.1.1, "Network Functions Virtualisation (NFV); Management and Orchestration", Dec. 2014

[5] ETSI GS NFV-IFA 005, "Network Function Virtualisation (NFV); Management and Orchestration; Or-Vi reference point – Interface and Information Model Specification", v2.1.1, Apr. 2016

[6] ETSI GS NFV-IFA 006, "Network Functions Virtualisation (NFV); Management and Orchestration; Vi-Vnfm reference point - Interface and Information Model Specification", V2.1.1, Apr. 2016

[7] Avino, G., Malinverno, M., Malandrino, F., Casetti, C. E., Chiasserini, C. F., Nardini, G., & Scarpina, S. (2017). "A Simulation-based Testbed for Vehicular Collision Detection". IEEE VNC, Turin, Nov. 2017