5GCroCo Barcelona Trial Site Results: Orchestration KPIs Measurements and Evaluation

Leonardo Lossi*, Pol Alemany†, Javier Fernandez Hidalgo‡, Francesca Moscatelli*, Ricard Vilalta†, Raul Munoz,†
Roshan Sedar† and Miguel Catalan-Cid‡
*Nextworks, Pisa, Italy
†Centre Tecnològic de Telecomunicacions de Catalunya (CTTC/CERCA), Castelldefels, Spain
‡i2CAT, Barcelona, Spain

***Abstract*—The 5GCroCo project has been working on aspects around vehicular communications taking into consideration different perspectives such as i) infrastructures with multiple administrative domains where to achieve the most efficient service deployment, and ii) the automotive services themselves for providing advanced functionalities to vehicular users. This article aims to present the results collected on top of the 5GCroCo Barcelona small-scale trial infrastructure during the experimental phases. In particular, the goal of the conducted experiments was to evaluate how the proposed orchestration solution works on top of a multi-domain infrastructure. Composite network slices, i.e., composed of multiple slice subnets and related network services (NSs) have been orchestrated and managed across the Barcelona multi-domain infrastructure and the different deployment iterations have been monitored to validate whether the Key Performance Indicators (KPIs) selected were accomplished.**

I. INTRODUCTION

5G technologies are driving to deep transformations and creating new business opportunities in the automotive industry. In particular, with the introduction of Connected, Cooperative and Automated Mobility (CCAM) services higher safety, higher traffic efficiency and lower environmental pollution will be ensured. To deliver CCAM services the automotive industry needs to collaborate with the Telecoms in order to provide vehicle-to-everything (V2X) communications, thus involving the seamless delivery of services with URLLC, which is required by typical real-time applications. Fulfilling these Quality of Service (QoS) requirements becomes uncertain and challenging in cross-border scenarios: crossing a geographical border usually means entering or resuming roaming status while changing the Mobile Network Operator (MNO), this makes difficult preserving the requested end-to-end QoS for applications running on different physical networks.

In this context arises the need of handling one of the most complex tasks in delivering CCAM services, the deployment of a complete V2X infrastructure. This task must face several standardization, regulatory and legal issues that involve many stakeholders (such the MNOs), transport authorities, road operators and service providers. Considering this multi-operator, multi-country, multi-car-manufacturer and multi-vendor scenario, the preservation of the QoS required by CCAM services becomes challenging [1]. The H2020 5GCroCo project [2] aims at defining successful strategies for CCAM services provisioning in cross-border scenarios to reduce the uncertainties of the future real 5G cross-borders deployments. This project foresees performing trials at small-scale to test the defined use cases and architectures in cross-border scenarios to be deployed afterwards in large-scale trials. The Barcelona small-scale site aims to trial the Anticipated Cooperative Collision Avoidance (ACCA) use case, which deals with the possibility to exploit an exchange of data between vehicles in order to anticipate the detection of dangerous events that typical stand-alone on-board sensing systems will not be able to detect. In addition, the ACCA application aims at localizing the dangerous events with a sufficient level of anticipation to induce a smoother and more homogeneous vehicle reaction.

This paper presents the results of a study on the 5GCroCo Orchestration Framework's KPIs related to the provisioning and the termination of the ACCA application across the complete V2X communications infrastructure in the Barcelona small-scale trial site. This paper is organized as follows. Section II presents the architecture of the Barcelona small-scale trial site and Section III describes the ACCA application architecture. Section IV introduces the adopted tools and the 5GCroCo guidelines for the experimentation methodology. Section V shows the experimentation architecture and an explanation of the defined KPIs being computed from the collected data. Section VI presents the experimentation results and finally Section VII concludes the paper.

II. BARCELONA SMALL-SCALE TRIAL SITE

In order to carry out experimental validations, the so-called Barcelona small-scale trial site was used as illustrated in Fig.1. This infrastructure makes use of a real world infrastructure deployed in two different geographical places. On one side, an emulated cross-border Internet Exchange Point (IXP) platform allocated in CTTC premises (Castelldefels). On the other hand, a 5G Neutral Hosting platform was deployed in the 22@ district in Barcelona, having a distance of 25 km between both sites. Regarding the communications between both sites, a dedicated Ethernet and optical transport network was used as part of the 5G Barcelona [3] infrastructure.

The 5G Neutral Hosting platform was designed with the capability to manage a set of virtual MNOs (vMNOs) deployment. By doing so, a cross-border entity was emulated in which each vMNO gives service to a single country. All the vMNO are designed to offer LTE connectivity for both scenario, vehicular environments and MEC infrastructure to offer services in the edge domain of the network infrastructure. On the other side, the cross-border IXP is used by vMNOs to exchange information among them as it offers the physical infrastructure. Moreover, a set of public cloud servers are

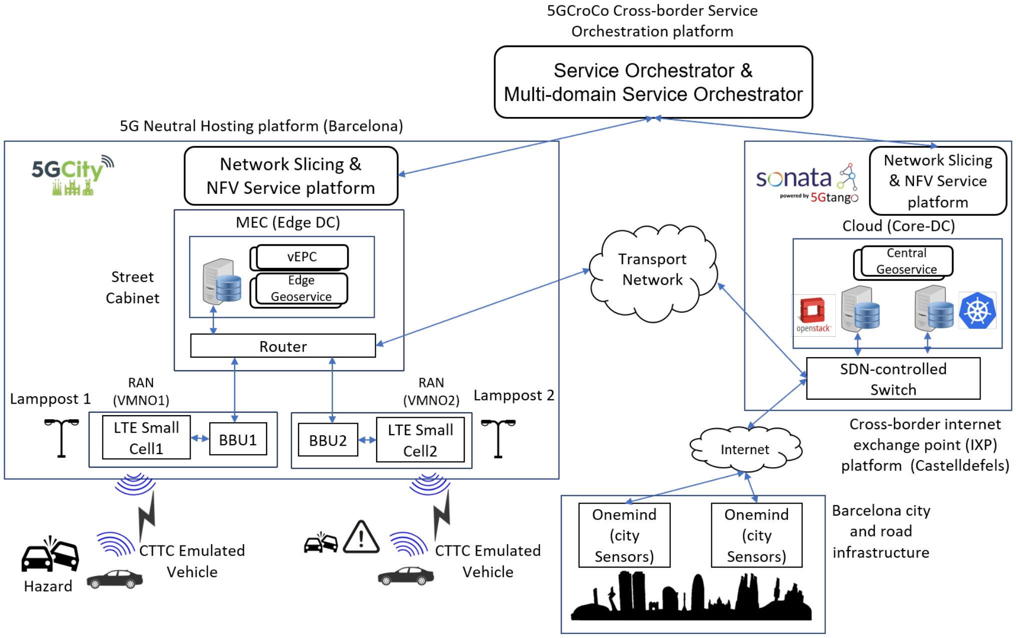

Fig. 1: Barcelona small-scale trial site architecture

available in order to offer cloud computing resources to the VMNOs.

A. 5G Neutral Hosting

The 5G Neutral Hosting solution tries to represent the way different vMNOs can share a common physical infrastructure [4], in this particular case provided by the municipality of Barcelona.These MNOs can still offer their proprietary Network Function Virtualization (NFV) NSs while running in isolation, by means of slicing techniques. The software implementing this solution is built as a stack of components. These components implement north and south bound interfaces which allow them to communicate and interact with their counterparts, realizing a micro services based architecture where each service taking care of a specific functionality. The Neutral Host platform is built based on the following components:

- **Slicing and Orchestration Engine (SOE)**: SOE is in charge of the Network Slices lifecycle management (LCM). In particular, it takes care of the management and partitioning of the infrastructure resources (compute, storage and networking resources) and the orchestration of the vertical services running there.
- **OpenStack (Virtual Infrastructure Manager - VIM) and RAN Controller**: SOE interacts with OpenStack and the RAN Controller, to secure and manage compute and radio resources, so that they can be associated with a particular slice. In our case we have two different Network Slices, mimicking two different domains.
- **ETSI Open Source MANO (OSM)**: Once resources are part of a slice, SOE will interact then with ETSI OSM (through the Multi-Tier Orchestrator for the case in which the system would have more than one NFV Orchestrator) to manage the NSs instantiated in that slice. Note that these NS instances are using the compute and radio resources managed by the OpenStack and RAN Controller.

B. Cross-border Internet Exchange Point

The cross-border IXP has two main functionalities. On one hand to manage and offer the physical infrastructure for the vMNOs to communicate among them and, on the other hand, to provide a set of public cloud resources for the vMNOs where to deploy services. The fact of making cloud resources public helps with the need in some applications to centralise the information generated by multiple sources that can not be exchanged directly like in our scenario, where different vMNOs are managing each its own MEC domain. Moreover, the cross-border IXP is able to manage deployed services based on two different technologies: Virtual Network Functions (VNFs) and Cloud-native network Functions (CNFs).

The cross-border IXP makes use of the internal CTTC testbed called ADRENALINE [5]. In order to have connectivity to exchange data between the multiple vMNOs and the cloud servers, a switch with Software Defined Networking (SDN) functionalities is implemented by using Open vSwitch on a commercial off-the-shelf (COTS) hardware. On a layer above, an OpenDaylight (ODL) based SDN controller is used to manage and control the connectivity between the 5G Barcelona transport network, the core datacenter (DC) and two container-based servers using Kubernetes (K8s) in the ADRENALINE testbed. Each of the two K8s servers has its own K8s controller, but the core-DC is composed of an internal packet-based network and a computing resources cluster. The internal packet-based network is composed of a set of OpenFlow switches which are controlled with ODL. Finally, the computing resources cluster is available through a multi-node OpenStack controller with one controller and two compute nodes.

Concerning the creation of the E2E connectivity across the physical infrastructure, the SDN controllers make use of the Transport API (T-API) [6] to have the context and topology information of each domain and request the best Connectivity Service (CS) possible when requested. Finally, and above, the ODL SDN controller, a NFV platform is implemented and it is responsible deploying the virtual services on the core-DC andor the two K8s servers and to requests the necessary CSs to the SDN Controller below. The NFV platform is based on the SONATA Service Platform (SP) [7] and it is able to manage the life-cycle of individual NS but also Network Slices (i.e., a Network Slice is composed by one or more NSs) until they need to be terminated.

C. Service Orchestrator and Multi-domain Service Orchestrator

In the Barcelona small-scale trial site, where a cross-border scenario is emulated, the ACCA application components are orchestrated across three different domains: a central public cloud managed by the IXP platform and two edge locations belonging to different vMNOs and managed through the 5G Neutral Hosting platform. This deployment implies a functional split of the application into slice subnets and NSs to be deployed and managed by different domain-specific platforms, thus requiring additional orchestration logic to perform a coherent E2E provisioning and LCM of the overall application. Such additional logic is provided by a centralized E2E cross-border Service Orchestration platform that is composed by two main functional blocks: the Service Orchestrator (SO) and the Multi-Domain Orchestrator (MSO). These two components, leverage the network slicing concept and implement a Network Slice data-model based on 3GPP TS 28.541 v17.1.0 specification [8], where each subnet hosts a nested NS running in a single domain.

More in detail, the SO, based on the Vertical Slicer prototype [9] [10], manages the deployment of the ACCA service components that have to be deployed in different domains. The SO identifies and converts the high-level requirements (e.g., number of supported vehicles) and the ACCA E2E service chain into a composite Network Slice Template (NST) customized for the ACCA QoS parameters. Then the SO requests the provisioning of the subnets composing the NST (one for each target domain) in order to cover all the required geographical areas.The MDO coordinates the provisioning of the composite Network Slice across the edge and core locations providing the logic to request the instantiation of the constituent subnets in each underlying domain. Then, the MDO coordinates the LCM of the E2E Network Slice across the domains-specific platforms (i.e., the IXP and the Neutral Hosting platforms). In order to perform these actions, the MDO provides domain-specific platform drivers; specifically, for the Barcelona small-scale trial, the MDO provides drivers for both the Neutral Hosting and the IXP platforms.

III. ACCA Application Architecture

Fig.2 shows the proposed cross-border ACCA application architecture deployed in the Barcelona small-scale trial. At the bottom, vehicles are connected to any of the two vMNOs deployed to emulate a cross-border scenario. The vehicles used in the Barcelona small-scale trial are equipped with a car PC system which can be used as Communications Control Unit (CCU) in any vehicle, and it can also be used as vehicle emulator. Two different protocols are considered for the communication of vehicles with the backend: UDP (User Datagram Protocol) carrying ETSI-ITS encoded messages including Basic Transport Protocol (BTP) and Geo Networking headers, and MQTT (Message Queuing Telemetry Transport) either also with ETSI-ITS payload as for UDP or with JSON encoded messages.

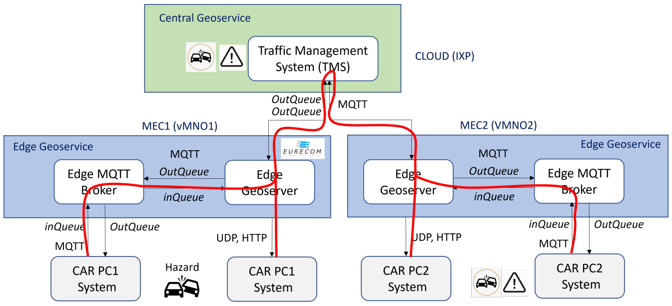

Fig. 2: Barcelona small-scale trial Cross-border ACCA architecture

Each vMNO deploys an Edge Geoservice to provide low latency ACCA Geoservices. The Edge Geoservice includes two applications, the Geoserver and the MQTT broker. The main task of the Geoserver is to gather awareness, perception and sensor information from vehicles and other IoT devices, consolidate them and evaluate potential hazards. The Geoserver manages directly hazard-related information from vehicles using UDP. It also connects to the Edge MQTT broker to synchronize hazard related data between different types of services (MQTT or UDP), and to the Central Geoservice to transmit and receive wider scope hazard-related data. The Edge MQTT broker is the functional entity that receives and sends messages to and from multiple vehicles using MQTT based on how such clients send (publish) and wish to receive (subscribe) over topics. MQTT is a lightweight, simple messaging protocol based on a publish/subscribe model, designed for constrained devices and low bandwidth overhead. Finally, a Central Geoservice is deployed in the IXP cloud for cross-border scenarios. It deploys the Traffic Management System (TMS) application. A TMS is a complex distributed application that enables the monitoring, management and operation of large infrastructures such as a road or a city. It allows the interconnection of heterogeneous systems and offers a central MQTT broker, which handles incoming packets from the different Geoservices deployed in the MEC and orchestrates such information to forward them to interested parties almost immediately. More detailed information on the cross-border ACCA architecture deployed the Barcelona trial is available in [11].

IV. Experimentation Methodology

The 5GCroCo monitoring guidelines foresee the usage of a monitoring platform to ensure an automated monitoring of the E2E system and its proper operation in real time. To this end, the monitoring platform collects data to perform an automated computation of KPIs and to produce their visualization for the final evaluation. According to the proposed methodology, each component in the 5GCroCo ecosystem should send logs (partially or fully) to a centralized log collector in order to compute and store KPIs, eventually aggregating metrics coming from different components. A common message string format is used across components and use cases to reduce the load and to simplify the parsing algorithms implemented in the log collector. This message format is composed by a list of whitespace-separated different mandatory elements: `<component> <type> <direction> <partner> <information> at <timestamp>`.

The 5GCroCo monitoring platform is based on the Elastic Stack [12] open source project components (i.e., Elasticsearch, Logstash, Kibana and Beats) that have been exploited to match the project's requirements related to components to be monitored and related KPIs. In the rest of this section we report briefly how the different components of the Elastic Stack have been used and integrated to realize the automated KPIs monitoring.

Beats are open source log shippers to be deployed for each entity to be monitored. In particular, during the Barcelona trial a Filebeat instance has been deployed in each Virtual Machine (VM) hosting the different orchestration platforms (i.e., IXP, Neutral Hosting and SO/MDO) in order to harvest the log files produced by each of them and to send messages, according to the specified format, related to the provisioning and the termination of the ACCA application's components. Logstash handles the streaming of a large amount of log data coming from different sources. In the Barcelona set-up, log data coming from Filebeat instances are processed to determine the time consumed to instantiate and terminate the ACCA application and the related E2E Network Slice in a cross-border scenario, where different administrative entities have to cooperate and exchange information. This component processes the log data using a pipeline-based workflow: i) data are first parsed into events, ii) the produced events are then filtered in order to perform an intermediary processing, where different filter plugins are combined and applied conditionally to produce new events, and iii) new events are processed to compute the KPIs of interest. Finally values produced by the Logstash implemented pipeline are shipped to the Elasticsearch time series data base, which is used to store, index and easily retrieve them through the available REST API. Finally, Kibana is a browser-based UI which allows searching, analyzing and visualizing the data stored inside Elasticsearch indices.

V. EXPERIMENTATION ARCHITECTURE AND KPIS DEFINITION

Figure 3 shows how the Elastic Stack components have been deployed across the Barcelona small-scale trial site. A Filebeat instance has been placed in each VM where orchestration platforms are deployed and has been configured to harvest line by line a proper log file to detect the relevant events by exploiting Regular Expressions (RegExps). The collected log lines are then passed to a processor script to produce events whose message is compliant with the common format detailed above. All the events produced by Filebeat instances are conveyed to a VM with running Logstash and Elasticsearch instances. Logstash has been configured to listen for events transmitted by Beats instances, to aggregate them and to compute the KPIs of interest, i.e., the ACCA application's provisioning and termination time. Finally, Logstash sends the final event containing the computed KPIs values to Elasticsearch on the same host to be easily and efficiently queryable.

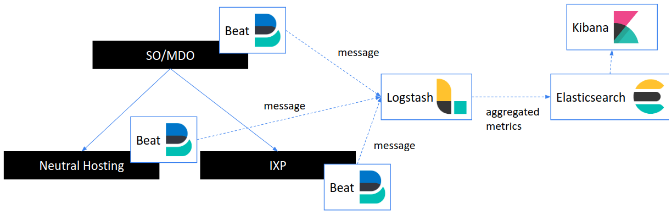

Fig. 3: Barcelona small-scale trial Elastic Stack deployment

In order to compute the KPIs, Logstash has been configured to run five pipelines and to perform the so-called multi-stage pipeline aggregation. The first stage pipeline is responsible for receiving and dispatching the events coming from the three involved orchestration platforms (i.e., IXP, Neutral Hosting and SO/MDO) to three different second stage pipelines. These second stage pipelines exploit the aggregate filter plugin to compute intermediate measures starting from multiple related events. The intermediate measures are finally conveyed to a sink pipeline where they are aggregated to compute the final KPIs values. The aggregation process on Logstash is configured to deal with out of order events processing, which may occur due to the different aggregation tasks that could introduce delays. The final event contains different KPIs values related to the provisioning and termination of the ACCA application. In the rest of the section we provide a brief description of the different values automatically computed through the multi-stage pipeline aggregation.

The *MDO instantiaton latency* represents the amount of time between the timestamps related to the start of the ACCA application instantiation event at the SO and the last request performed by the MDO for triggering the instantiation of related subnets at the uderlying domains. Then, it consists in the time consumed at the SO/MDO to process an incoming request for the provisioning of the ACCA application, to translate the high-level requirements into a NST and to request the instantiation of the constituent subnets to the single-domain platforms.

The *total instantiation latency* consists in the total latency perceived at the SO/MDO for performing the ACCA application provisioning. This value is computed using the timestamp related to the start of the instantiation and the timestamp related to the end of the subnets instantiation. The SO/MDO considers the procedure successfully executed when it detects that all the different subnets composing the E2E Network Slice are properly instantiated and configured. The detection of the subnets current status is performed by executing a polling procedure with a 60 seconds frequency. This implementation choice makes this KPI less focused on the real amount of time needed to deploy and configure the ACCA E2E application. Indeed the computed value includes the time consumed to pool the status of the different subnets, which may be detected as configured with a maximum delay of 60 seconds.

Then the *provisioning time* KPI has been introduced and it is computed as the sum of the *MDO instantiaton latency* and the maximum subnet instantiation latency directly provided by the IXP and the Neutral Hosting platforms. Indeed, at the IXP ad Neutral Hosting the different requests for subnets provisioning are handled in parallel and the corresponding latencies are computed as the difference between the subnet instantiation start event and the related end event. This small shrewdness allows to cut out the time related to the polling procedure performed at the SO/MDO, thus resulting in a more precise estimation of the time needed to perform the resource provisioning for the ACCA application cross-border deployment.

The KPIs related to the termination and so to the release of the resources employed in the ACCA application deployment are specular with respect to the instantiation ones. The *MDO termination latency* is the amount of time between the timestamps related to the start of the ACCA application termination event and the last related request to terminate subnets at the single-domain platforms. This value represents the ACCA application termination processing time at SO/MDO. The *total termination latency* is the latency perceived at the SO/MDO with respect to the overall ACCA application termination. As in the previous case the procedure is considered as successfully performed when all the subnets are detect as terminated, performing a polling procedure with a 60 seconds frequency. Finally the termination time KPI is computed as the sum of the *MDO termination latency* and the maximum subnet termination latency provided by the IXP or the Neutral Hosting platforms, cutting out the time related to the polling procedure performed by the SO/MDO to compute a more precise estimation of the time needed to release resources allocated during the ACCA application cross-border deployment.

VI. EXPERIMENTATION RESULTS

The objective of this experimentation is to perform an analysis to evaluate the Orchestration Framework performances in terms of provisioning and termination time for the ACCA application. In particular, this study focuses on the deployment of the ACCA components across the three described domains administrated through the IXP and Neutral Hosting platforms. The ACCA E2E service is composed of three subnets, each of them hosting a NS. At the edge, the two Edge Geoservice NSs are both composed of a single VNF that includes the Geoserver and the MQTT broker. At the public cloud, the Central Geoservice NS has a single VNF hosting the TMS. The experimentation has been conducted using two types of images: the first type, referred in this paper as ”uncompressed image”, is simply a snapshot of a running instance of each VNF, the second one, referred instead as ”compressed image”, is a compressed version of the first one. The compression methodology consists on the conversion of the uncompressed image to a qcow2 compressed image. Table I reports the relevant information for each VNF to be deployed.

The graph shown in Figure 4 shows the provisioning time across 50 executions of the instantiation procedure using the uncompressed images. Table II reports the aggregated metrics for the same 50 executions with the two types of images. The reported values for the *provisioning time* KPI state that the time consumed for the ACCA application provisioning, excluding the polling time, is near 3 minutes in average for both the compressed and uncompressed images, then main benefits of using compressed images are related to the saving of disk space for accommodating multiple provisioning requests for multiple vMNOs. Indeed, the behaviour of the specific VIM (i.e., OpenStack) foresees to uncompress the images before deploying them, introducing an overhead that can be observed through the values reported in Table II. In the same way, we can observe that the termination time is around 1,4 minutes for both cases. Going more in detail, Table III and Figure 5 report the partial times collected at the single platforms, highlighting which are the time components with the highest impact on the overall service provisioning and termination. The collected data allow to understand the impact of the processing performed at each orchestration platform on the provisioning and termination time. The peculiarity of the 5GCroCo Orchestration Framework is the presence of a centralized cross-border coordination logic provided by the SO/MDO. According to the trial results, the time consumed by the SO/MDO for coordinating the ACCA application E2E provisioning, i.e., 19 seconds in average, is not impacting the overall time consumed for deploying the constituent subnets. Looking at the provisioning time, the introduced overhead is not significant with respect to the actual time used to deploy the VNFs at the single-domain platforms, which is in average 2,5 minutes for the IXP and 0,7 minutes for the Neutral Hosting, and that is strictly dependent on the application sizing and compute infrastructure capabilities.

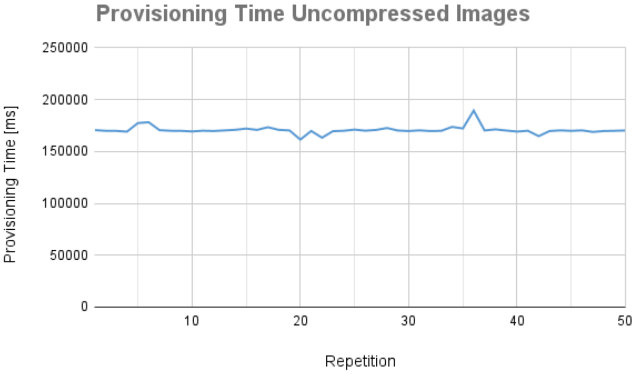

Fig. 4: Provisioning time using uncompressed images in milliseconds

According to the 5GPPP phase II KPIs report [13], the overall service creation and provisioning time for a single-domain deployment should be lower than 90 minutes, taking into account also the time consumed for the on-boarding of descriptors/images to the Orchestration Framework. The 5GCroCo proposed solution is capable of achieving a provisioning time of nearly 3 minutes, thus satisfying a common usual target of 5 minutes. Taking into consideration the application sizing, the trial infrastructure and the fact that this measure includes also the time consumed for coordinating the deployment among the different domains, the evaluated performances in terms of provisioning time can be considered as a valid result.

TABLE I: Geoservice VNFs

	Image Size [GB]		**Deployment Flavour**		
	Compressed	**Uncompressed**	**CPUs**	**RAM [GB]**	**Storage [GB]**
Edge Geoservice	17	47	2	8	60
Central Geoservice	14.67	55.37	2	8	150

TABLE II: Barcelona small-scale trial results in milliseconds

	Uncompressed Images			**Compressed Images**		
	Max	**Min**	**Mean**	**Max**	**Min**	**Mean**
Provisioning Time [ms]	189435	161475	170638.74	210057	179975	188172.88
Termination Time [ms]	110062	68159	84318.34	105061	68574	82514.92

TABLE III: Provisioning time components in milliseconds (uncompressed images)

	Mean		
	SO/MDO	**IXP**	**Neutral Hosting**
Provisioning Time [ms]	19059.9	151579.24	41294.92

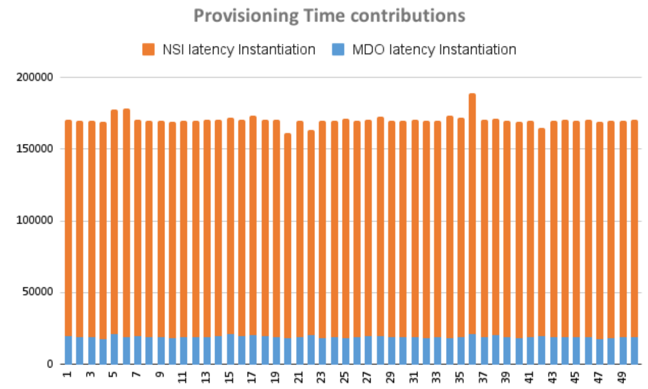

Fig. 5: Contributions of MDO latency and subnets latency to the provisioning time (uncompressed images)

VII. CONCLUSIONS

The 5GCroCo Orchestration Framework leads to tangible benefits with respect to the automation of LCM procedures by considerably reducing the complexity of handling the ACCA application provisioning, configuration and termination in a cross-border scenario, where multiple administrative entities have to interact and exchange information to achieve the proper E2E service operation. This study also states that the introduction of a centralized architecture component, i.e., the SO/MDO, for handling the coordination of the deployment in a coherent manner, is not introducing significant delays with respect to the time consumed for performing the subnets provisioning and termination. Indeed, as observed through the collected data, the coordination overhead represents a tiny part of the overall measured latency, where the most part of the consumed time remains related to the spawing/termination of the VNFs and the related images at the single-domain platforms. These values are mostly dependent on the application sizing and infrastructure capabilities in terms of available compute resources and thus may vary and can be further improved as NFV infrastructures evolve. Taking into consideration a usual common target of 5 minutes for a single-domain deployment, the achieved results are considered optimal given the ACCA application sizing and the Barcelona small-scale trial infrastructure.

ACKNOWLEDGMENT

Work partially supported by the EC H2020 5GCroCo (No825050) and the ”Grant RTI2018-099178-B-I00 funded by MCIN/AEI/ 10.13039/501100011033 and by ERDF A way of making Europe”.

REFERENCES

[1] A. Kousaridas and et al., “5g cross-border operation for connected and automated mobility: Challenges and solutions,” *Future Internet*, vol. 12, no. 1, p. 5, 2020.

[2] “H2020 5gcroco - fifth generation cross-border control,” 2022. [Online]. Available: https://5gcroco.eu/

[3] “5g barcelona end-to-end infrastructure,” https://5gbarcelona.org/, accessed: 2022-01-21.

[4] A. Fernández-Fernández and et al., “Validating a 5g-enabled neutral host framework in city-wide deployments,” *Sensors*, vol. 21, no. 23, 2021. [Online]. Available: https://www.mdpi.com/1424-8220/21/23/8103

[5] R. Muñoz and et al., “The ADRENALINE Testbed: An SDN/NFV Packet/Optical Transport Network and Edge/Core Cloud Platform for End-to-End 5G and IoT Services,” in *2017 European Conference on Networks and Communications (EuCNC), IEEE*, Jun. 2017.

[6] V. Lopez and et al., “Transport api: A solution for sdn in carriers networks,” in *ECOC 2016*, 2016.

[7] “Sonata SP,” https://sonata-nfv.eu/, accessed: 2022-01-21.

[8] “Ts 28.541 v17.1.0 management and orchestration; 5g network resource model (nrm); stage 2 and stage 3,” 2021. [Online]. Available: https://www.3gpp.org/specifications

[9] C. Casetti and et al., “The vertical slicer: Verticals’ entry point to 5g networks,” in *2018 European Conference on Networks and Communications (EuCNC)*. IEEE, 2018.

[10] “Vertical slicer,” 2022. [Online]. Available: https://github.com/nextworks-it/slicer

[11] R. Muñoz and et al., “5gcroco barcelona trial site for cross-border anticipated cooperative collision avoidance,” in *2020 European Conference on Networks and Communications (EuCNC)*. IEEE, 2020, pp. 34–39.

[12] “Elastic stack,” 2022. [Online]. Available: https://www.elastic.co/elastic-stack/

[13] 5G PPP, “5g ppp annual progress monitoring report,” 2019. [Online]. Available: https://5g-ppp.eu/wp-content/uploads/2020/10/5G-PPP-PMR2019v1-6.pdf