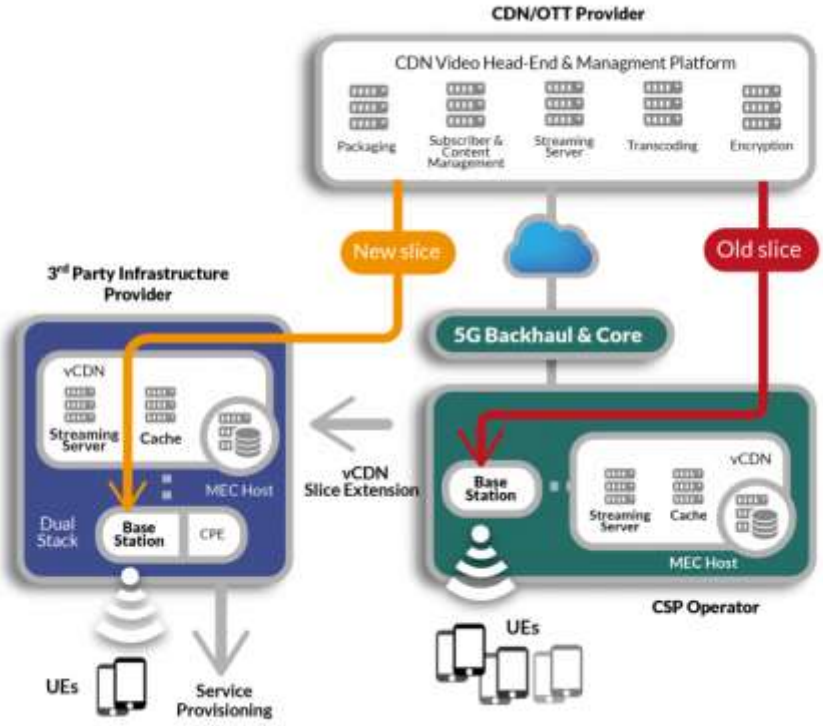
5GZORRO Cloud-Native Open Source MEC Platform

5GZORRO architecture is created to enable several advanced multi-provider 5G use cases. One of the use cases, Pervasive vCDN Services put forward by INTRACOM, is depicted in the figure below. The use case calls for on-demand provisioning of additional service resources based on user requests load, so that the additional resources can belong to a 3rd party resource provider. As a result, the original Network Slice provisioned for the service is extended to span more than one resource provider in a way transparent to service users. Such seamless Network Slice extension relies on quite a few capabilities of the 5GZORRO architecture that deserve separate discussion each. In this post, we focus on one specific technology enabler created by the project team, namely the 5GZORRO Multi-access Edge Computing (MEC) Platform shown in the use case figure as "MEC Host".



We stop to briefly recall the related terminology. ETSI GS MEC 001 (link) defines the glossary of terms around the MEC technology and defines *MEC System as* a collection of MEC hosts and MEC management necessary to run MEC applications*; MEC Host* as an entity that contains a MEC platform and a virtualisation infrastructure which provides compute, storage and network resources to MEC applications; *MEC platform* as a collection of functionality that is required to run MEC applications on a specific MEC host virtualisation infrastructure and to enable them to provide and consume MEC services, and that can provide itself a number of MEC services; and *MEC Management as a* MEC system level management, i.e. the collection of management components which have the overview of the complete MEC system, and MEC host level management, i.e. the collection of components which handle the management of the MEC specific functionality of a particular MEC platform, MEC host and the MEC applications running on it.

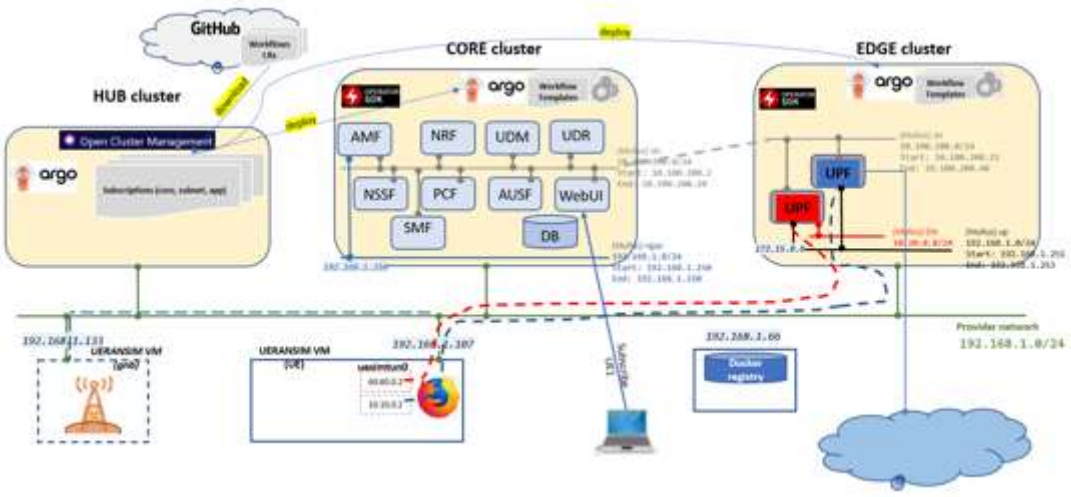
The key idea behind 5GZORRO MEC platform realization is embracing the cloud native paradigm as fully as possible. First, we containerize MEC service components to be deployed on MEC platform as

CNFs. In addition, we employ Kubernetes (k8s), both as a MEC Host and as a MEC Management, running and orchestrating the MEC-hosted CNFs. This way, cloud native experience spans both the data plane and the control plane of the resulting MEC platform, complementing the overall 5GZORRO cloud native approach to 5G Orchestration.

In 5GZORRO, Cloud Native MEC Platform (CNMP) is a per-domain component, meaning that it is assumed to be installed in all the participating resource providers. Developed as a separate component with clear interfaces, our CNMP implementation allows future integration of other MEC platforms into 5GZORRO platform. The main idea is to use the same K8s clusters to run both the NFVI components, e.g., 5G Core functions, as well as application and service components and manage them as a uniform cloud native environment. Thus, to instantiate a slice or a service, the system first creates a declarative specification of this slice or service, including all the required control plane and data plane components, and then delivers it for deployment to the right K8s cluster(s). In k8s, the declarative specification for custom deployable artefact is created as Custom Resource Definition, or CRD. 5GZORRO CNMP implementation extends a multi-cluster k8s environment with custom CRDs describing the 5G Slice and Service definitions and with custom workflows implementing 5G Slice and Service Lifecycle.

To realize the envisioned MEC Platform, we had to use a softwarised implementation of 5G Core. We have found the open source free5GC project to be the most advanced with respect to the implemented 5G functionality. On the other hand, we have found that the implementation is not following the intent-based approach of cloud native software and relies on inflexible configuration mechanisms. To put it short, free5GC is a solid "dockerized" implementation of 5G Core Service Based Architecture which can be run on k8s as a bunch of separate containers under external orchestrator. In 5GZORRO, we have "kubernetized" free5GC implementation to make it deployable as fully cloud native micro-service application that can be natively orchestrated by k8s with no need in additional external orchestrators. In addition, we developed novel 5G Core and UPF Operators that act as VNFM for these CNFs with Argo Workflows acting as NFV Orchestrator.

Using the "kubernetized" free5GC, as well as the developed CRDs, Operators, and Workflows, we have enabled the use case presented in the beginning of this post, namely dynamic slice extension to the required edge location based on user load. As shown in figure below, when the system experiences increased load near the supported edge location, it automatically dispatches the required 5GCore and Service components to the edge provider's MEC platform and allows users to consume the service locally enjoying better experience.



This capability was demonstrated by 5GZORRO team at EUCNC'22; implementation and operational prototype can be found in https://github.com/5GZORRO/issm-mec-cnmp.