# Experiments with Large-scale Lightweight Service Slices (ELSA)

## 1.1. Scope of this document

This document is to describe the purpose of the demonstration and the architecture of the test infrastructure that was used by the NECOS consortium. This test infrastructure is not precluding that any other can be used to run de demonstration. The guide to install the software in the substrate resources is provided in the README file, in the same repository as the software.

## 1.2. Introduction

This demonstration is composed by one slice provider hosted by UFPA, the Tenant to make the requests hosted by UFRN, and the resource providers hosted at UCL, as presented in **Figure 1**.

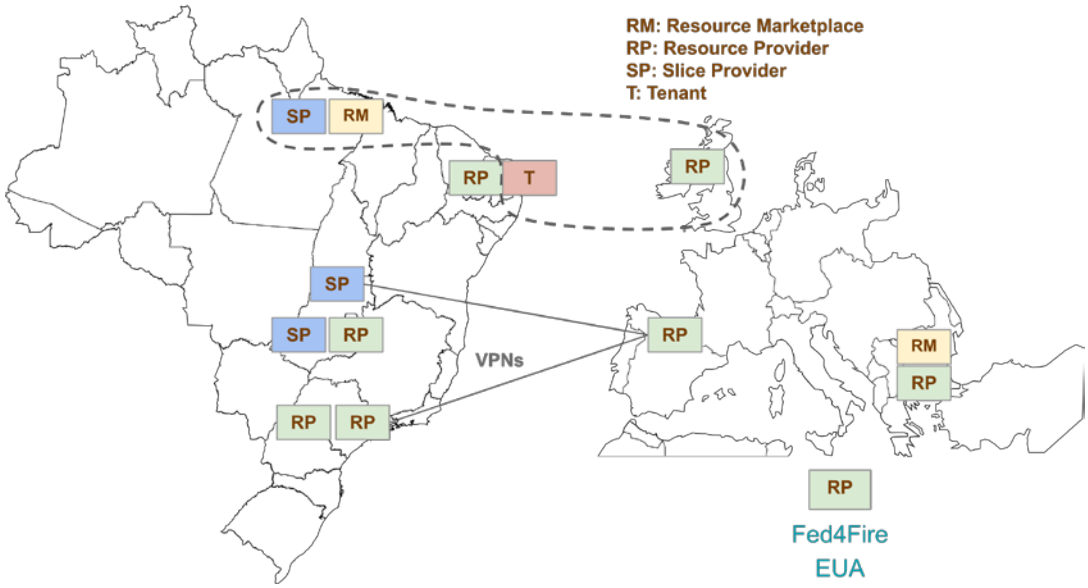


**Figure 1.** Instantiation of ELSA on the experimental infrastructure.

## 1.3. License

All the source code developed within the NECOS project, and made available as OSS, is released under the Apache License – Version 2.0[1]

[1] http://www.apache.org/licenses/LICENSE-2.0

## 1.4. Objectives

The idea of this demo is to show the deployment of end-to-end Slices that will be utilised by a Tenant in order to host services consisting of a very large number of lightweight elements (i.e., Virtual Network Functions (VNFs) and vLinks) deployed at the Edge of the infrastructure. We will demonstrate how the Tenant is able to reuse their existing software components by attaching them to the allocated end-to-end slice in a completely transparent way. The description of the desired end-to-end Slice is provided as YAML input by a Slice Activator component in the Tenant domain to a software component that implements some of the functionalities of both the NECOS Slice Builder and the Slice Resource Orchestrator.

The descriptor will include information (i.e., type of VIM, size, etc.) about the slice parts to be created, the links between the slice parts and the monitoring parameters (KPIs) to be collected from each of them. In this demonstration, the marketplace-related workflows will not be considered, and the YAML descriptor provided to the Slice Builder will contain a predefined set of entry-points of the Slice Controllers to be contacted in the Resource Providers.

The end-to-end Slice specified via the above descriptor will automatically be built on a Slice Provider hosted at UFPA. At the UCL premises, 12 of the available 14 interconnected physical servers will be hosting different instances of the DC Slice Controller, in order to emulate different NECOS Resource Providers. The created DC slice parts will be based on the on-demand instantiation of the Very Lightweight Network & Service Platform (VLSP) VIM, which will support the creation of simple lightweight service topologies across the different slice parts of an end-to-end Slice (mainly simple video streaming services).

In order to orchestrate the deployment of the above mentioned large-scale services on an end-to-end Slice, an instance of the open source (5GEx) ESCAPE[2] Orchestrator will be configured to use the resources of that end-to-end Slice as substrate for embedding the required service elements (i.e., VNFs and virtual Links). This will happen transparently as the Tenant will attach their existing service orchestrator (ESCAPE) to the newly created slices.

Finally, as soon as the end-to-end Slices will be up and running, the NECOS IMA will start collecting relevant KPIs related to it. The implementation of the DC Slice Controller deployed at the UCL Resource Provider is based on the instantiation of bare-metal slices. As such, the above collected measurements related to the KPI-9 (Physical Resource Utilization) will highlight how the execution of the large-scale services deployed on one of the slices will not affect the physical resources of the other slices in the same Resource Provider.

## 1.5. Workflow

The Experiment Controller component implementing some of the functionalities expected for the NECOS Builder and SRO architectural components will ensure large-scale system operations while the slices are created, operated and monitored.

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

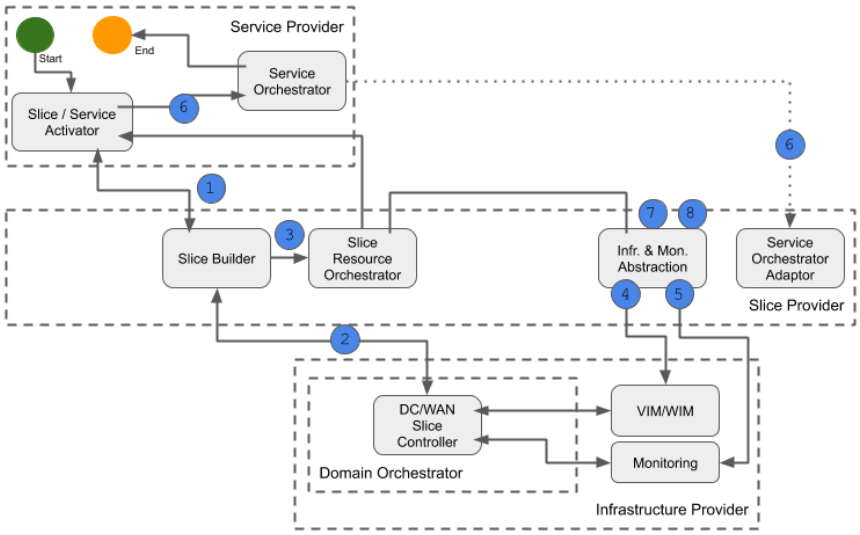


**Figure 2.** ELSA workflow.

More specifically the demo shows the following different steps (see **Figure 2**):

- **Step 1:** The YAML end-to-end Slice Specification (according to the NECOS information model) is provided by the Slice / Service Activator component in the Tenant domain to the software module implementing the required Slice Builder functions, in order to start the instantiation of a new end-to-end slice;
- **Step 2:** The above Slice Builder module interacts with different DC Slice Controllers instances that are already deployed in the testbed (via the Slice Instantiation Interface), in order to request the creation of different DC slice parts, each based on an on-demand instance of the VLSP VIM;
- **Step 3:** The information about the different allocated DC slice parts is returned back from the Slice Builder to the component implementing the functions of the Slice Resource Orchestrator. The latter will take care of interconnecting the allocated slice parts via creating an emulated tunnelling that will be based on the interaction of custom instantiated VLSP edge routers;
- **Step 4:** Resource Adapters attached to the allocated VIM endpoints are dynamically created and the handlers to the adapters are provided back to the Slice Resource Orchestrator;
- **Step 5:** Monitoring Adapters are requested to the IMA according to the allocated type of VIM and Monitoring Subsystem that were deployed in each slice part in order to gather monitoring data in a uniform way;
- **Step 6:** An instance of the open source ESCAPE service orchestrator is attached to the newly deployed end-to-end Slice and the Service Activator will receive a handle to the northbound interface of that Service Orchestrator instance. This will be utilised to request the ‘embedding’ of a large service request (in terms of number of involved service virtual elements) on the previously created end-to-end slice, which will act as the resource substrate. The orchestrator will also act as

Service Orchestrator Adapter as service requests will be translated in a format supported by the Slice Provider;

- **Step 7:** IMA collects and aggregates data (via an augmented Data Aggregator) from the different slice parts in order to generate KPIs related to the end-to-end Slice;
- **Step 8:** measurements related to the KPI-9 collected from IMA in the previous step that are related to (at least) two end-to-end Slice in the UCL Resource Provider will be considered. The bare-metal slices built by the DC Slice Controllers will guarantee resources isolation at the physical layer. As such, deploying a large-scale service instance on one of the slices, will not affect the utilization of the physical resources of the other ones.