Small cEllS coordinAtion for Multi-tenancy and Edge services

Grant Agreement No.671596

Topic: H2020-2014-ICT-14
Advanced 5G Network Infrastructure for the Future Internet
Research and Innovation Action

Deliverable D7.4

PoC Evaluation, KPI Assessment and Overall Evaluation

Document Number:	H2020-5GPPP-GA No.671596/*WP7/D7.4/21.12.2017*
Contractual Date of Delivery:	*31.12.2017*
Editor:	Alan Whitehead – ip.access Ltd. (IPA)
Work-package:	*WP7*
Distribution / Type:	Public (PU) / Report (R)
Version:	1.0
Total Number of Pages:	46
File:	*SESAME_Deliverable 7.4_v1.0_Final*

Abstract

This document assesses the overall success of the SESAME Proof of Concept demonstrator and how it "meets" the project's identified KPIs.

Version History

Version	Date	Comments, Changes, Status	Authors, contributors, reviewers
0.1	02.11.2017	Initial draft by IPA	A.G. Whitehead
0.2	05.12.2017	Contributions by ATOS	E. Jimeno
0.3	06.12.2017	Updated draft by IPA	A.G. Whitehead
0.4	11.12.2017	Updated draft by IPA	A.G. Whitehead
0.5	13.12.2017	Updated draft by IPA	A.G. Whitehead
0.6	15.12.2017	Contributions by NCSRD	I. Giannoulakis
0.7	19.12.2017	Updated draft by IPA	A.G. Whitehead
0.8	19.12.2017	Pre-final version by IPA	A.G. Whitehead
0.9	20.12.2017	Review by SMNET	A. Dardamanis
0.10	20.12.2017	Review by OTE	I. Chochliouros
1.0	20.12.2017	Final version by IPA	A.G. Whitehead
1.0	21.12.2017	Document submitted to the Commission	I. Chochliouros

Contributors

First Name	Last Name	Partner	Email
Alan	Whitehead	IPA	*alan.whitehead@ipaccess.com*
Elisa	Jimeno	ATOS	*elisa.jimeno@atos.net*
Ioannis	Giannoulakis	NCSRD	*giannoul@iit.demokritos.gr*
Athanassios	Dardamanis	SMNET	*adardamanis@smartnet.gr*
Ioannis	Chochliouros	OTE	*Ichochliouros@oteresearch.gr*

Glossary

Acronym	Explanation
3GPP	Third Generation Partnership Project
4G	Fourth Generation of Mobile Communications
5G	Fifth Generation of Mobile Communications
ARM	Advanced RISC Machine
BF	Broadband Forum
BS	Base Station
CESC	Cloud Enabled Small Cell
CESCM	CESC Manager
CM	Configuration Management
CN	Core Network
CP	Control Plane
CPE	Customer Premises Equipment
CPU	Central Processing Unit
DB	Database, Data Base
DC	Data Centre
DL	Downlink
DP	Data Plane
DPI	Deep Packet Inspection
ECMA	European Computer Manufacturers Association
EMS	Element Management System
EPC	Evolved Packet Core
E-UTRA	Evolved UMTS Terrestrial Radio Access
FP	Function Provider
FPGA	Field Programmable Gate Array
FQDN	Fully Qualified Domain Name
FW	FireWall
GA	Grant Agreement
GB	Giga Bytes
GbE	Gigabit Ethernet
GHz	Giga Hertz
GPRS	General Packet Radio Service
GTP	GPRS Tunnelling Protocol
GTP-U	GPRS Tunnelling Protocol User-Plane
GW	Gateway
H2020	Horizon 2020
HTTP, http	HyperText Transfer Protocol
HW	Hardware
IA	Innovation Action
IaaS	Infrastructure as a Service
ICT	Information and Communication Technology
ID, id	Identifier
IP	Internet Protocol
IRP	Integration Reference Point
IS	Information Service
ITU	International Telecommunication Union
JSON	JavaScript Object Notation
KHz	Kilo Hertz
KPI	Key Performance Indicator
LAN	Local Area Network
Light DC	Light Data Centre

LTE	Long Term Evolution
µs	micro-server
Mbps	Mega-bits per second
MCC	Mobile Country Code
MIB	Management Information Base
MME	Mobility Management Entity
MNC	Mobile Network Code
MOCN	Multi-Operator Core Network
NFV	Network Functions Virtualization
NMS	Network Management System
NO	Network Operator
NOS	Network Orchestration System
NS	Network Service
NSD	Network Services Descriptor
OS	Operating System
PC	Personal Computer
PHY	Physical Layer
PLMN	Public Land Mobile Network
PM	Performance Management
PM	Physical Machine
PNF	Physical Network Function
PoC	Proof of Concept
PoP	Point of Presence
PPP	Public-Private Partnership
PS	Power Supply
QCI	QoS Class Identifier
QoE	Quality of Experience
QoS	Quality of Service
R/W, r/w	Read/Write
RAN	Radio Access Network
RAM	Random Access Memory
RIA	Research and Innovation Action
RRC	Radio Resource Control
RRM	Radio Resources Management
SC	Small Cell
SC-C-VNF	Small Cell-Common-VNF
SCNO	Small Cell Network Operator
SFC	Service Function Chaining
SIB	System Information Block
SLA	Service Level Agreement
SOAP	Simple Object Access Protocol
SP	Service Provider
SS	Solution Set
SW	Software
TAC	Tracking Area Code
TEID	Tunnel Endpoint Identifier
TR	Technical Report
TS	Technical Specification
TU	Transcoding Unit
UC	Use Case
UE	User Equipment
UL	Uplink
UMTS	Universal Mobile Telecommunications System
UP	User Plane
UTRAN	UMTS Terrestrial Radio Access Network

VIM	Virtualised Infrastructure Manager
VM	Virtual Machine
VNF	Virtual Network Function
VNFM	Virtual Network Function Manager
VNFO	Virtual Network Function Orchestrator
VSCN	Virtual Small Cell Network
VSCNO	Virtual Small Cell Network Operator
vTU	virtual Transcoding Unit
VTU	Video Transcoding Unit
WAN	Wide Area Network
WP	Work Package
XML	eXtensible Markup Language

Table of Contents

List of Figures

List of Tables

1 Introduction

1.1 Deliverable outline

This document provides the report on the SESAME integrated pilot systems, their interconnection, use cases implemented and the evaluation results.

It describes the deployment of the pilot sites, including the equipment to complement the developed SESAME modules.

1.2 Definitions of Terms and SESAME concepts

2 Proof of Concept Demo

The SESAME Proof of Concept (PoC) demo is intended to demonstrate the feasibility of the following key concepts:

2.1 MOCN

Multi-Operator Core Network (MOCN) (see [1] and [2]) is a standard 3GPP technology that allows up to six network operators to "share" the radio spectrum of a host network operator (NO).
The radio base station (BS) broadcasts a list of up to six PLMN IDs (MCC and MNC) that it supports. The respective UEs select the PLMN they wish to access and communicate to the base station as part of the RRC Connection Setup procedure. The base station is then responsible for routing the UE's signalling and user plane traffic to the appropriate core network.

The SESAME PoC Demo is intended to support all of the functionality of standard MOCN.
However, in line with the SESAME architecture described in Deliverable D.2.5 [3] and illustrated by Figure 1 below, this functionality is both extended significantly and distributed across the components of the CESC in order to deliver the SESAME concepts detailed below.

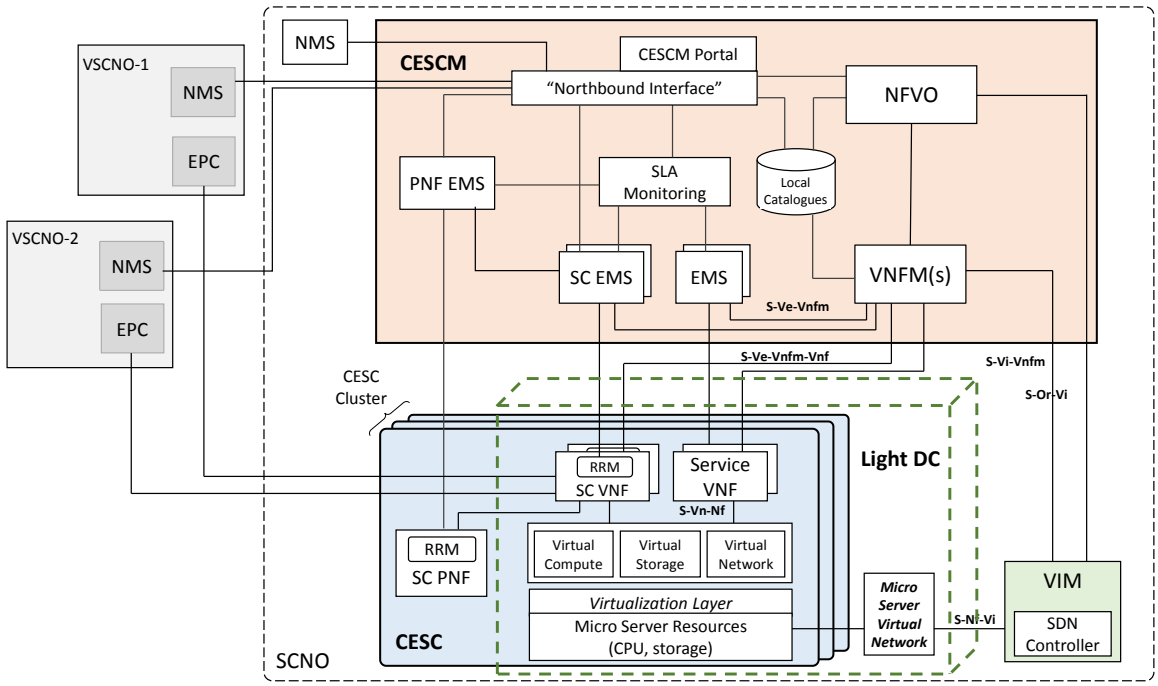

Figure 1 – SESAME Architecture and Component Interactions

2.2 Network Isolation

Network Isolation is a "key" SESAME concept. In standard MOCN, each participating network operator shares the RAN of the host network operator on a *first-come, first-served* basis and it is possible for one operator to consume more than their fair share of resources to the detriment of the other operators.

Network isolation helps to address this:

- Each Virtual Small Cell Network Operator is supported by a dedicated SC VNF running on the CESC.
- Each SC VNF handles only the S1 signalling and user plane traffic for the supported VSCNO.
- Each SC VNF runs in a dedicated virtual machine that provides networking, code and CPU resource isolation. Specifically:
  - IP traffic between the SC VNF and the VSCNO's EPC can be completely isolated from that of other VSCNO's as it is presented on a dedicated Ethernet interface[1]. Backhaul congestion caused by one VSCNO need not impact the others.
  - Each VSCNO can implement a distinct service chain that has no impact on the user plane traffic of other VSCNO. This also provides for service differentiation between VSCNOs.
  - The CPU resources (both RAM and processor) consumed by each SC VNF have no impact on the other SC VNFs hosted by the CESC as they run in separate virtual machines. In addition, it is possible to dimension resources available to each SC VNF according to the size of VSCNO's network slice.

[1] In practice, there is likely to be some element of common backhaul between the CESC and each operator's EPC.

The only exception to this scheme is the SC Common VNF that provides the S1 multiplexing and fan-out functionality. The S1 signalling traffic of each VSCNO hosted on a CESC passes through the SC Common VNF, which is responsible for routing it to and from the SC PNF and the appropriate SC VNF. However, this signalling traffic is a relatively small percentage of the overall load placed on the CESC and it is the user plane traffic, routed directly between the SC PNF and each SC VNF, which accounts for the bulk of the load.

2.3 Per-VSCNO Network Slice

2.3.1 Resource Caps

Unlike standard MOCN, which is an uncontrolled best-effort RAN sharing mechanism, SESAME provides the concept of a "network slice" where each VSCNO is allocated a defined portion of the resources available on a CESC as follows:

- A cap on the number of active UEs that each VSCNO may have at any one time.
- A cap on the maximum uplink bandwidth that each VSCNO may consume.
- A cap on the maximum downlink bandwidth that each VSCNO may consume.

The intention is that each VSCNO may be configured with different limits, as required, and that these limits can be changed on-the-fly, in response to events such as SLA violations.

2.3.2 Service Chaining

Service chaining provides a mechanism by which the user plane traffic of a VSCNO may be routed through a number of value added network edge services. As described in deliverable D7.2 [6], example service chain functions might include:

- Deep Packet Inspection (vDpi) ,
- Video Watermarking (vWatermark),
- Video Transcoding Unit (vTU),
- Firewall functionality (vFirewall),
- Video Analytics (vAnalytics),
- Content Caching (vCache).

A key aspect of the SESAME architecture is that each VSCNO hosted on a CESC may have a different service chain that is tailored to their specific needs and which may provide a degree of service differentiation.

2.4 SLA Monitoring

Emerging models in the cloud environment, so called SaaS (Software as a Service), makes software available that is hosted in different locations or on different underlying architectures. This new development model makes the evaluation of compliance terms from which a service is expected, to be a

sort of real challenge. The assessment of the quality of the service(s) (QoS) provided is critical for both parties in this context, (i.e.: for clients and providers), in order to report any eventual non-compliance of the terms agreed in a comprehensive and trusted manner.
The terms under which a SaaS application needs to be provided are defined by a Service Level Agreement (SLA). This approach describes how the specific service can be monetized, based on the performance and analytics from the service monitoring and the tier level contracted.

Current monitoring systems are restricted to a static and homogenous environment. In the case of SESAME, specific software has been developed in order to evaluate metrics of different natures, providing useful information for effective and efficient resource consumption management, supporting the decision-making process. This process is evaluated at runtime and allows the addition of new requirements or modification to existing ones, without stopping the service execution.

The mechanism to support the formal definition and management of the SLA relationship between the service provider and the stakeholder is defined in a lifecycle of several steps:

- Definition of the SLA Template specification. This procedure describes the SLA template in order to define the correct service description. There are some fields that could be modified in the negotiation process.

- Publication and Discovery, The Function Provider (FP) publishes a new template when a new VNF is uploaded to the platform where it is stored together with the rest of the metadata. Then the Service Provider (SP) publishes the catalogue with the different offers related to the services.

- Negotiation. The customer selects the predefined offer from the business catalogue and the trading process is formalised in the second step, once the service has been acquired.

- Resource selection, based on the selected SLA. The orchestrator allocates the resources for the assigned services.

- Monitoring and evaluation. The framework retrieves the metrics from the monitoring tools and compares them with all the terms of the SLA agreement.

- Accounting. The system informs when an SLA has been violated in order to open discussion between provider and customer to agree on a penalty due to the failure.

Before starting all the SLA processes, we have to assure the correct functioning of the procedures. For this purpose, the SLA framework is embedded with a testing tool in order to assure the software execution, based on a test preconditions and also a diagnosis is performed to assure the behaviour is as expected. This process is performed before the software is deployed, so that to ensure that the logic of the application is followed.
This methodology allows further development of features without the need to perform manual tests or to repeat the functionalities previously developed, and more importantly, that regression tests are introduced.

The following Figure shows the output produced by the SLA framework in the deployment process for the Demokritos' (NCSRD) use case. The framework can be tested individually, in case where some of the results of the components might have failed. These individual tests provide more specific information of the possible errors produced, and a log file is created with all the relevant information.

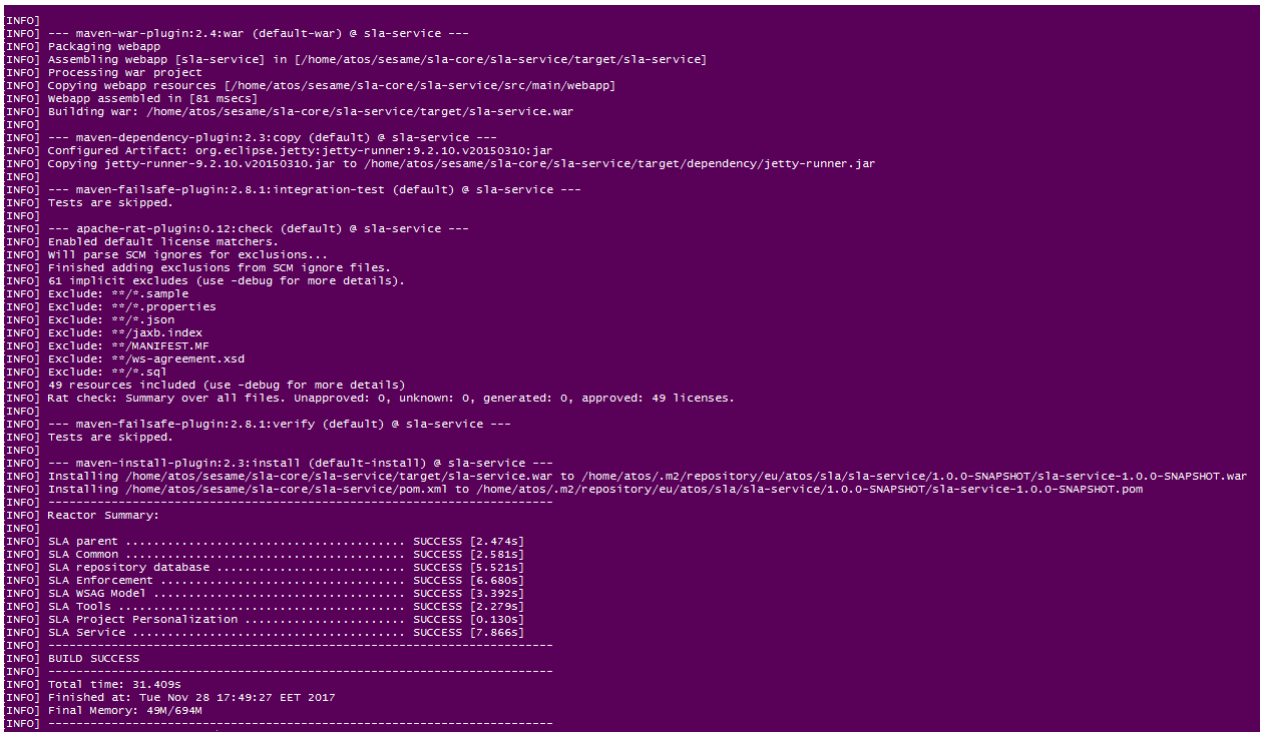

Figure 2 – SLA Framework Log

After the verification of the correct functioning of the software, and the communication with the rest of modules, it is then possible to integrate and start the SLA process of the service acquired.
The following part describes, in more detail, the results produced by the different steps mentioned above, including: Definition of the SLA Template specification; Publication and Discovery; Negotiation; Resource selection, *and*; Monitoring and Accounting.

The SLA template definition is created in a way based on Network Services Descriptor (NSD). After the creation it is forwarded to the management module, an internal verification is performed once the template is received, to check that there was no error in the process.

The publication of the services is presented in the CESCM Portal as a list of available services to be contracted in the platform.
The user will have to select the required service from this user interface, when creating a new Network Service (NS).

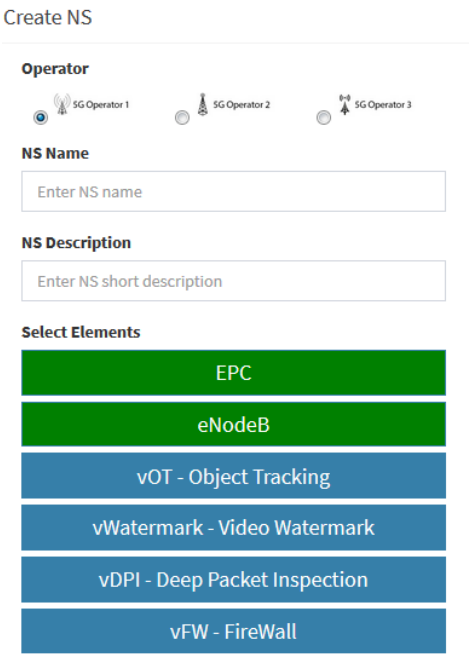

Figure 3 – Network Service Selection

The agreement is the result from the negotiation phase where the customer selects the predefined offer from the business catalogue and the trading process is formalised in the second step, once the service has been acquired.

The selection of the resources is defined by a flavour, which is defined independently of the kind of service.
In this sense, it is aligned with what is understood in the virtualisation environment as a deployment flavour. The generic framework for the definition is described as follows:

Flavour	Properties
Gold	- Highest Priority Service - Network traffic QoS with highest priority - Access to the IT resources should be prioritised
Silver	- Statistical Prioritisation for the service - Limitation on features (IT resources, Network resources, …). - Guarantee the minimum requirements in terms of resources
Bronze	- Equal to a Best Effort service but with an asterisk - More restricted limitation on features (IT resources, Network resources, …). - The system guarantees the IT resources required for the service to be operational.

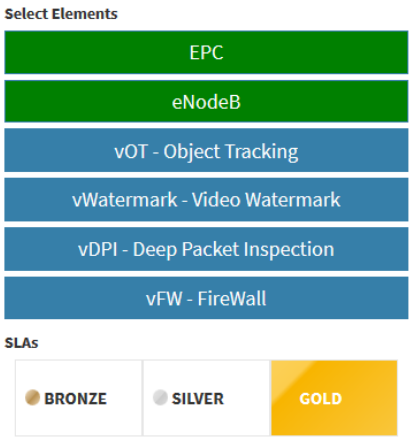

Figure 4 – Example "Flavour" definitions

Enforcement is the process by which the selected flavours agreed between both parties are evaluated in order to "see" if the provider complies with the agreement terms. The external monitoring tool, independent from the SLA module, retrieves the metrics needed to enforce an agreement and performs the assessment. The periodic process is shown below:

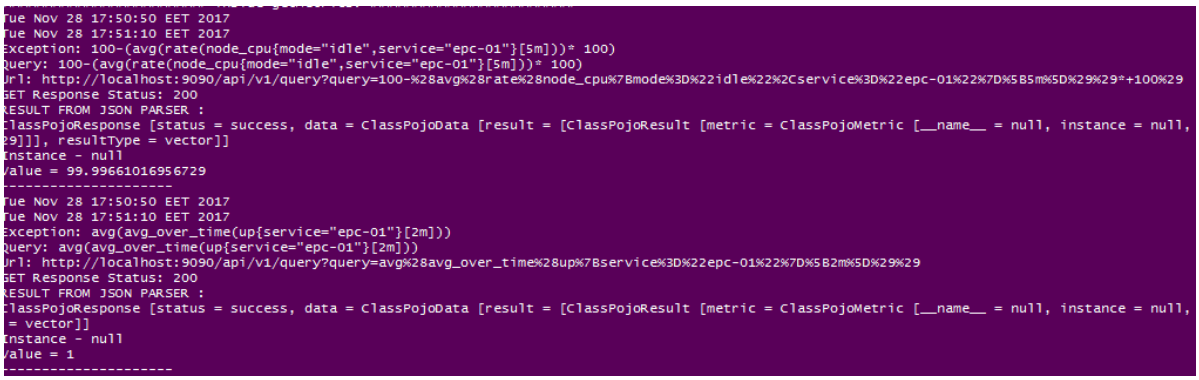

Figure 5 – SLA Monitoring Process

It has been implemented as a business rule, from which a Business Value can be raised in case any element do not satisfy a guaranteed term defined.
This information is also notified to the user, in order to monitor the status of the selected service.

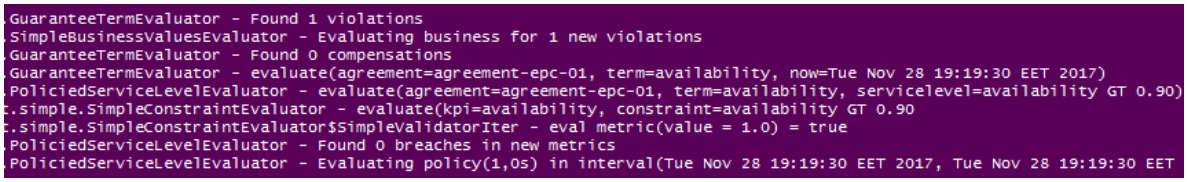

Figure 6 – SLA Evaluation Results

2.5 CESCM –EMS SLA Integration

Integration between the CESC Manager (CESCM) and the EMS allows the CESCM to collect performance management data from the EMS and present it as a consolidated view in an appropriate dashboard, raise alerts and, *where possible*, take corrective action.

In the SESAME PoC, a consolidated EMS based on the ip.access Network Orchestration System (NOS) [7] provides both the VNF EMS and PNF EMS functionality illustrated in Figure 1.
It receives performance management reports from the PNFs that conform to the format specified in 3GPP 32.435 [8], computes the corresponding KPI values and presents this information on a northbound interface[2] for access by the CESCM.

[2] For further details also see, *for example*: https://en.wikipedia.org/wiki/Northbound_interface.

3 Use Case Scenarios

3.1 Multi-Tenancy

3.1.1 Virtual Small Cell Network Operators (VSCNOs)

The VSCNO is a concept implemented primarily by the SESAME EMS that enables the participating operator to create and manage a network of virtual cells (their virtual network or network slice) in a manner that is "analogous" to managing a real, physical, network.

The main aspects of this concept are provided by a hierarchy of managed objects representing the VSCNO's virtual network.
Each VSCNO is able to view and interact with their own network slice but is unable to view and interact with the objects belonging to another VSCNO. The EMS object hierarchy is illustrated by Figure 7, below.

The objects with a pink background are created by the Small Cell Network Operator (SCNO) when a new VSCNO is enrolled. The objects with a green background are created by the VSCNO as part of their virtual network provisioning.

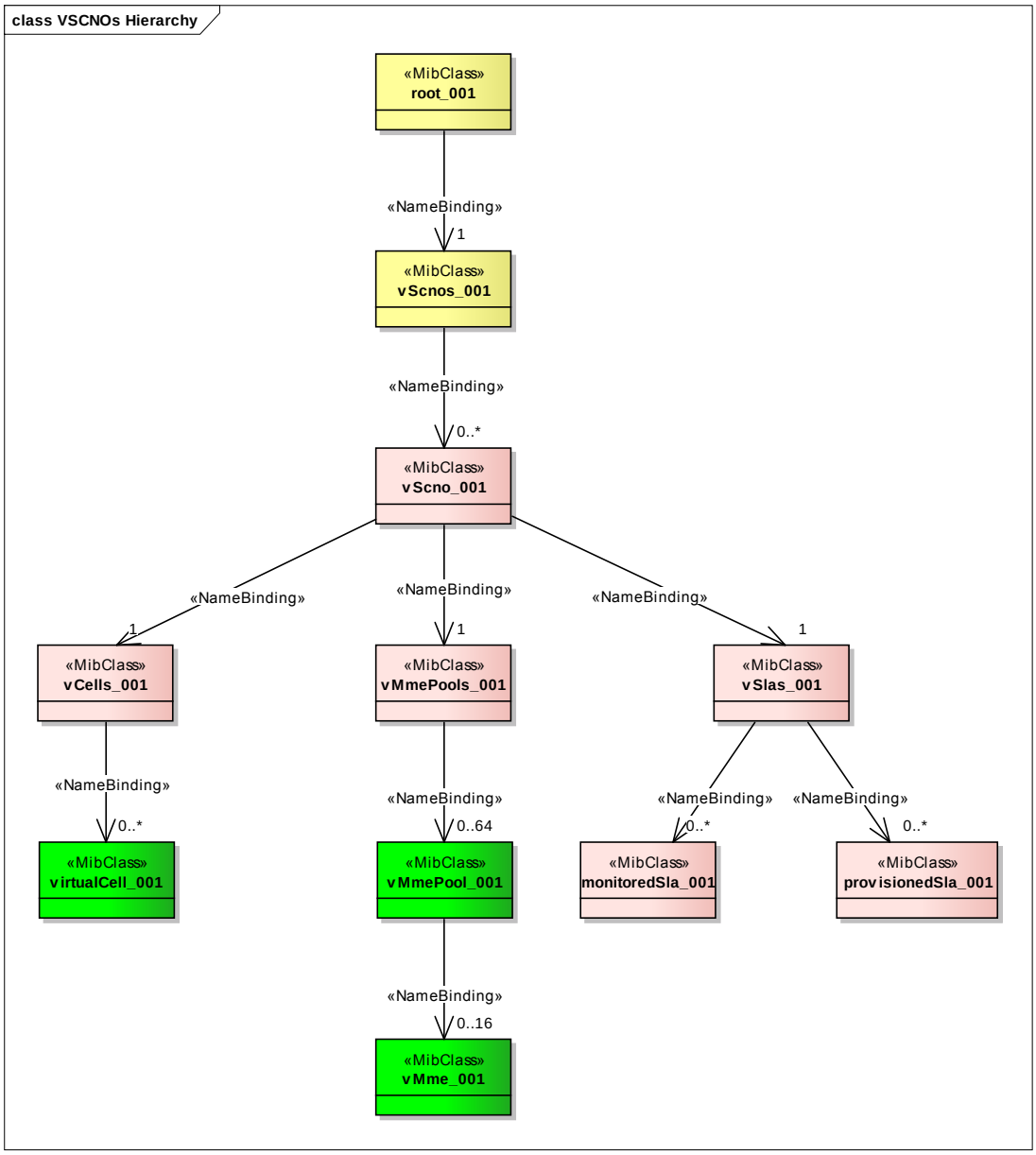

Figure 7 – VSCNOs object hierarchy

3.1.1.1 Root Object

The object represents the root of the System MIB presented by the EMS. It has no configurable attributes. Its child objects represent the many different functions managed by the EMS and, in Figure 7 as above, the sub-tree representing Virtual Small Cell Network Operators (VSCNOs) is illustrated.

3.1.1.2 VSCNOS Object

This is a top-level object, visible to the Small Cell Network Operator (SCNO) that owns the physical network infrastructure. It contains all of the child VSCNO sub-trees. It has no configurable attributes.

3.1.1.3 VSCNO Object

This is a top-level object that defines the root of a particular VSCNO's virtual network. The attributes of this object include:

- The VSNO's PLMN ID.
- A free form text field that may be used to describe the VSCNO.
- A link to the default SLA for use when provisioning a virtual cell (see 3.1.1.8 below).
- The VSCNO's user ID for remote access to the EMS.

3.1.1.4 MME Pools Collection Object

This is a collection object beneath which MME pool objects may be created. It has no configurable attributes.

3.1.1.5 MME Pool Object

An MME pool is a collection of MMEs that serve a set of LTE tracking areas. Together, the MMEs that are members of the pool that serves each specified tracking area in its entirety.
The attributes of this object specify the MME Group ID shared by MMEs in the pool and the list of served Tracking Area Codes (TACs). Each child MME object represents an MME belonging to the pool.

3.1.1.6 MME Object

Each MME object represents a single MME in the VSCNO's EPC[3]. Its attributes include its MME code, which must be unique within the pool, and its endpoint address (either an FQDN or an IP address).
When provisioning a new virtual cell, the VSCNO selects the MME, as represented by this object, which will be used by the SC VNF to terminate the virtual cell's S1 traffic.

Note that the PNF supports only a single TAC[4] and that, *therefore*, all virtual cells hosted by a physical cell share the same TAC. Thus, the selected MME object must support the TAC of the PNF associated with the selected CESC.

3.1.1.7 SLAs Collection Object

This is a collection object beneath which Provisioned and Monitored SLA objects are created by the SCNO on behalf of the VSCNO. It has no configurable attributes.

[3] A VSCNO may not, *in fact*, have their own EPC but they require access to one in order to terminate their traffic. Thus, where a physical network owner provides traffic terminating services to multiple VSCNOs, these VSCNOs may each create identical MME objects within their separate virtual network sub-trees.

[4] This restriction is common to all small cells that use the TR-069 data model [13].

3.1.1.8 Provisioned SLA Object

The Provisioned SLA object is created by the SCNO to capture the configuration management aspects of a Service Level Agreement reached between the SCNO and the VSCNO.
It is selected by the VSCNO as part of the virtual cell creation process to specify the limits of a network slice in terms of the maximum number of UEs with the VSCNO's PLMN that may be active on the virtual cell at any one time and the maximum uplink and downlink throughput in Mbits/s that these UEs can consume.

The Management Information Base (MIB) permits the SCNO to create multiple instances of Provisioned SLA objects with the intention that these form a menu of SLA options, from which the VSCNO may choose when provisioning a new virtual cell.

Testing has demonstrated that multiple object instances may be created, that the VSCNO may select one during the virtual cell provisioning process and that the EMS correctly propagates the parameters of the SLA to the appropriate SC-VNF.

3.1.1.9 Monitored SLA Object

One or more Monitored SLA objects are created by the SCNO to "capture" the performance management aspects of a Service Level Agreement reached between the SCNO and the VSCNO.
The configuration of each such object has three dimensions:

- The geographic scope that defines which virtual cells are covered by the SLA. These may be specified as all of the cells belonging to the VSCNO, a specific list of cells or all of the cells hosted on CESCs within a particular geographic region. The PoC demo implements only the first two of these options.
- The temporal scope that defines when the SLA is applied.
- The associated KPIs and their thresholds.

As PM counters are received by the EMS, it computes each defined KPI and updates its database (DB) with the results. These results are then assessed by an SLA Monitoring process that inspects each Monitored SLA object, in order to determine whether the configured KPIs have been met and, if not, what action to take.

The ability for the SCNO to create multiple Monitored SLA objects provides significant flexibility. They can, for example:

- Specify a default SLA that applies to all the virtual cells of a VSCNO.
- Specify site specific SLAs that cover a specific geographic area, list of cells or a specific time-frame. For example, the temporal scope of cells in a shopping mall would define only those hours that the mall is open.
- Break SLAs into groups such as "Availability", "Accessibility" and "Retainability"[5].

3.1.1.10 Virtual Cells Collection Object

This is a collection object beneath which the VSCNO creates Virtual Cell objects by means of the EMS Create Virtual Cell wizard. It has no configurable attributes.

[5] These are standard terms used by mobile operators to categorise performance measurements.

3.1.1.11 Virtual Cell Object

Each Virtual Cell object represents the VSCNO's network slice on a specific CESC and provides the VSCNO with the ability to manage their virtual cells in a manner that is analogous to the management of a physical cell.

For example:

- The EMS configuration management view allows the VCSNO to view their virtual cells and associated properties via a graphical user interface.
- The VSCNO is able to provision and de-provision virtual cells as desired without reference to the SCNO.
- In a commercial deployment, the *Operational State*[6] and *Availability Status*[7] parameters of the Virtual Cell would reflect its ability to provide service, based on whether -or not- the associated PNF was transmitting and whether -or not- the associated SC-VNF was operational, with an S1 connection to the VSCNO's EPC. This option has not been implemented in the PoC demo.
- The VSCNO is able to temporarily take a virtual cell out of service by administratively locking it. Locking a virtual cell causes the VSCNO's PLMN to be marked as "Reserved for Operator Use" on the host PNF, effectively taking it out of service. This option is fully supported by the PoC demo.

3.1.2 Per-VSCNO PM Reports

The SESAME PNF, as provided by the ip.access E40, has been enhanced to report a number of performance management counters on a *per-PLMN* basis.
The following per PLMN counters have been defined:

Counter	Description
GTP-U Packets Received per PLMN without Sequence Number	Provides the total number of downlink GTP-U packets received with the Sequence Number Flag (s) unset, indexed by PLMN and per QCI.
GTP-U Packets Received per PLMN with Sequence Number	Provides the total number of downlink GTP-U packets received with the Sequence Number Flag(s) set, indexed by PLMN and per QCI.
GTP-U Packets Transmitted per PLMN	Provides the total number of uplink GTP-U packets transmitted, indexed per PLMN and per QCI.
GTP-U Octets Received per PLMN	Provides the total number of downlink GTP-U octets received, indexed per PLMN and per QCI.
GTP-U Octets Transmitted per PLMN	Provides the total number of uplink GTP-U octets transmitted, indexed per PLMN and per QCI.
RRC Connection Establishment Success per PLMN	Provides the number of successful RRC establishments for each PLMN and establishment cause.
Maximum RRC Connection Set-up Time per PLMN	Provides the maximum time per PLMN, per establishment cause it takes to establish an RRC connection.
Mean RRC Connection Set-up Time per PLMN	Provides the mean time per establishment cause it takes to establish an RRC connection per PLMN.

[6] Operational State is a standard managed object property, defined in [3], that indicates the ability of the managed object to provided service.
[7] Availability Status is a standard MO property, defined in [3] that enlarges on the value of Operational State.

Counter	Description
Maximum RRC Connections per PLMN	Provides the maximum number of RRC Connections per PLMN during each granularity period.
Mean RRC Connections per PLMN	Provides the mean number of RRC Connections per PLMN during each granularity period.
RRC Connection Re-Establishment Attempts per PLMN	Provides the number of RRC connection re-establishment attempts per PLMN for each re-establishment cause as defined in [32.425].
Failed RRC Connection Re-Establishments per PLMN per Re-Establishment Cause	Provides the number of RRC re-establishment failures per PLMN for each re-establishment cause as defined in [32.425].
RRC Connection Re-Establishment Success per PLMN	Provides the number of successful RRC connection re-establishments per PLMN for each re-establishment cause as defined in [32.425].

Table 1 – Per-PLMN PM Counters

All of these counters are fully supported by the PoC demo. To date, testing has mainly focused on the GTP-U throughput counters as part of the VNF throughput capping feature.

3.1.3 Per-VSCNO KPIs

As described in section 3.1.1.9, per VSCNO KPIs are provided by instances of the Monitored SLA managed object.
The attributes of these objects define which KPIs should be monitored, their associated thresholds and the action to take on SLA breach. The PoC demo supports only the raise alarm action but in a commercial deployment the supported actions could be extended to include:

- Scaling up or down the computing resources assigned to the associated VNF.
- Changing the relative size of the network slice assigned to the VSCNO or possibly other VSCNOs on the same CESC.

3.1.4 EMS SLA Monitoring

The EMS SLA Monitoring service assesses each Monitored SLA object on a periodic basis and compares the most recent KPI values against the configured thresholds.

If a threshold breach is detected, a configuration option allows the EMS to raise an alarm on the affected Monitored SLA object. As managed objects with outstanding alarms are colour coded by the EMS, its Fault Management view allows a VSCNO to quickly form a view as to whether their SLAs are being met.

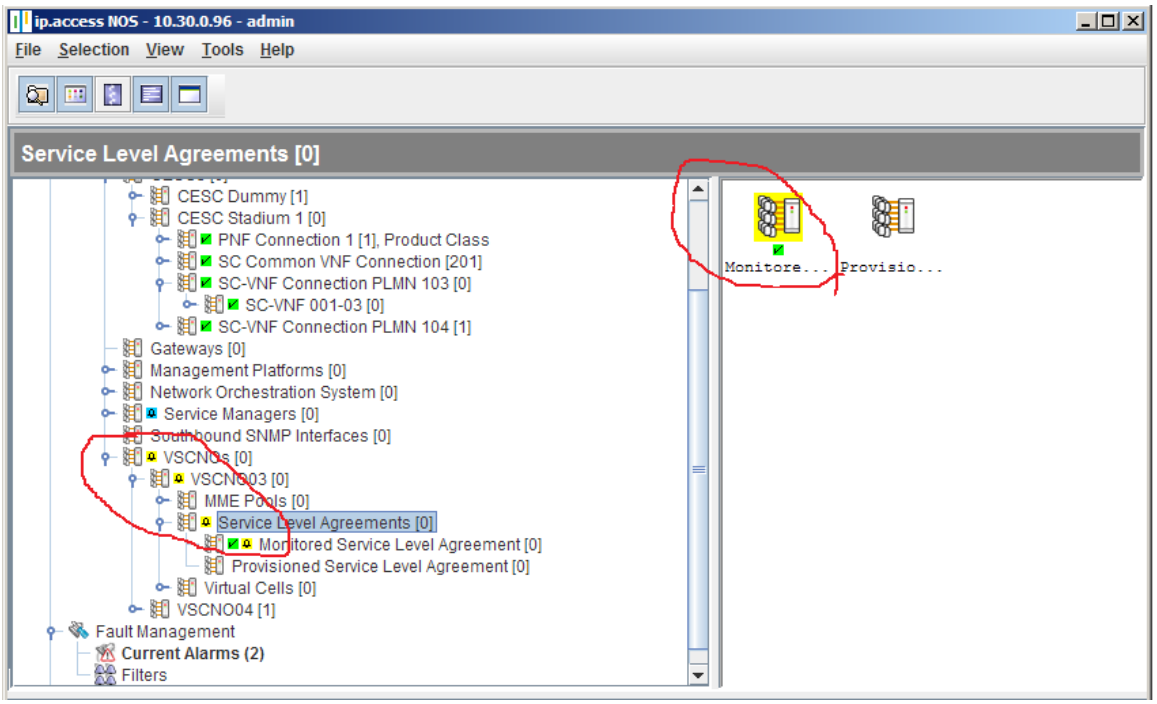

Figure 8 – Example Monitored SLA object with outstanding SLA Breach Alarm

The "bell" symbols in the left-hand pane indicate an outstanding alarm.
The EMS propagates this visual status upwards, so each managed object displays a status that indicates the highest severity alarm condition of itself and all its child objects.
In the right-hand pane, the background of the icon representing the Monitored SLA object is coloured yellow to represent an outstanding alarm of severity minor.

In Fault Management View, the alarm details are shown as illustrated in Figure 9 below. The top right pane shows the list of recent alarms and the selected entry is the "SLA Breach Detected" alarm.
The bottom left pane shows the list of notification associated with the selected alarm and their time of occurrence.
Currently, there is only one notification for the SLA breach alarm. The bottom middle pane shows the details of the notification whilst the bottom right pane provides help about the selected alarm.

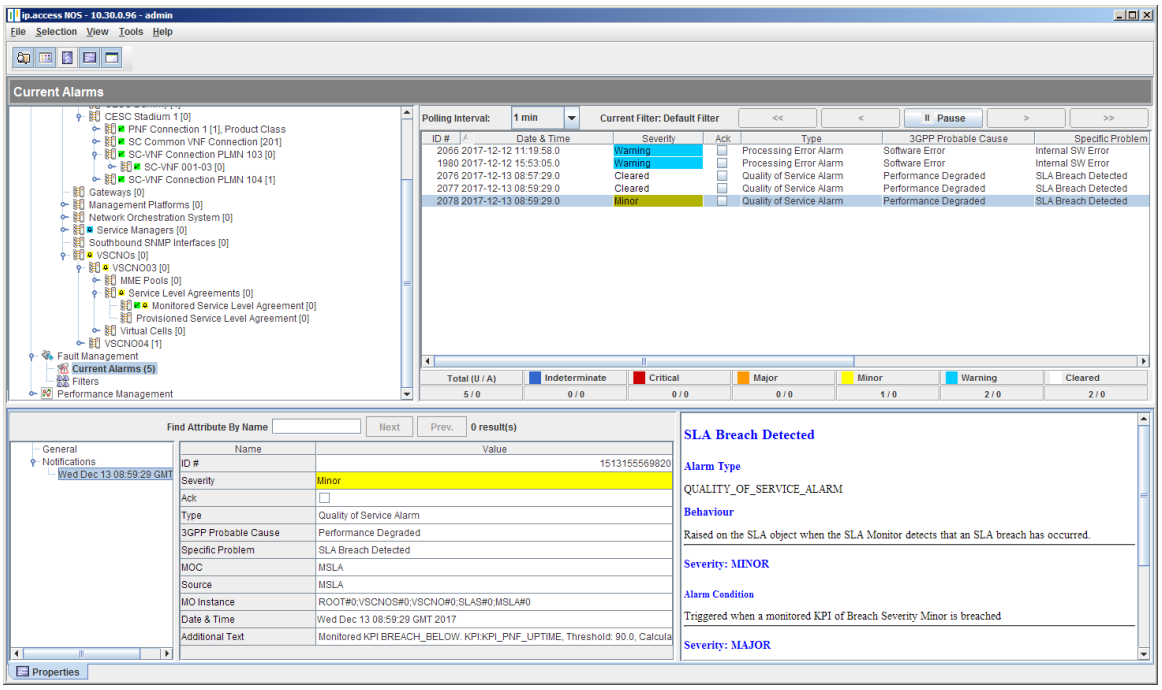

Figure 9 – SLA Breach Alarm Details

3.1.5 User Permissions

A described in section 2.2, a significant aspect of multi-tenancy is network isolation; as far as possible, the network slice of each tenant VSCNO is isolated from that of other VSCNOs. This concept continues through the EMS.

The EMS supports multiple simultaneous users and the set of managed objects that an individual user is able to view and interact with is controlled by user permissions.
As the network infrastructure owner, the SCNO is able to enrol VSCNO users and assign their permissions such that each VSCNO only has access to those managed objects that represent their individual network slice (see the Pink and Green shaded objects in Figure 7). They have no access to the managed objects of other VSCNOs or those of the SCNO.

The EMS provides a useful permissions group feature that allows the SCNO to define a set of permissions for each VSNO and then apply them in a simple step when enrolling a new user. Figure 10, below, shows the Group Permissions dialogue box where the user is assigning the permissions for VSCNO003 with PLMN ID “001,03”. As illustrated, this group allows full access to the managed object sub-tree of VSCNO03 and no access to the sub-tree of VSCNO04.

Figure 11 shows how the group defined in Figure 10 is used to assign permissions to a user.

Figure 12, Figure 13 and Figure 14 respectively show:

- The SCNO’s view that includes all VSNOs.
- VSCNO03’s view that shows only their network slice.
- VSCNO04’s view that, similarly, shows only their network slice.

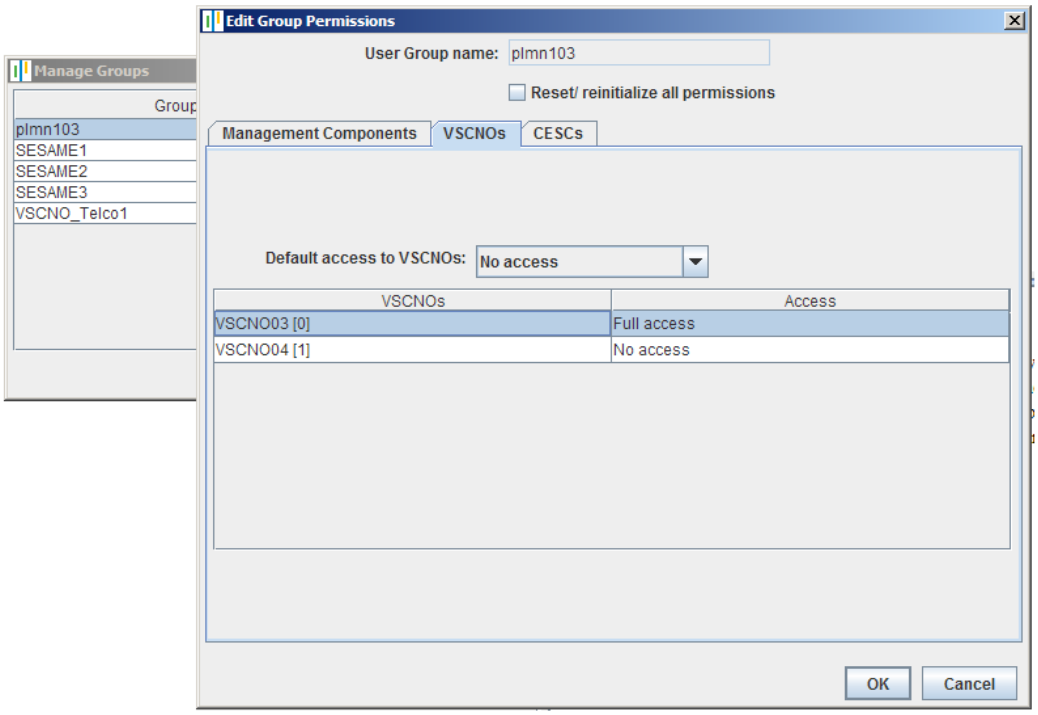

Figure 10 – EMS Group Permissions

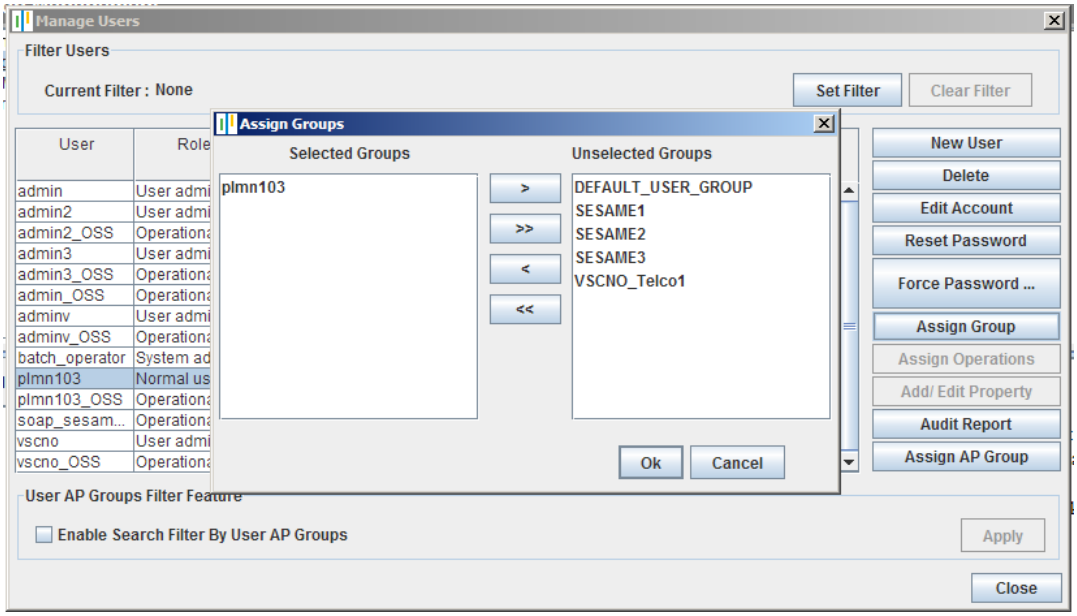

Figure 11 – Assigning Group Permissions to a User

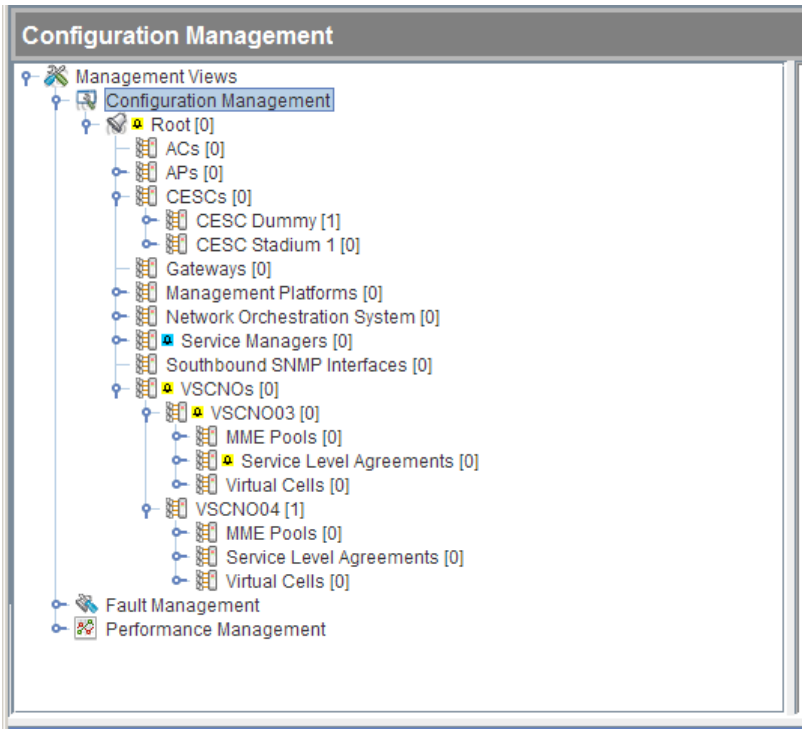

Figure 12 – SCNO Configuration Management view

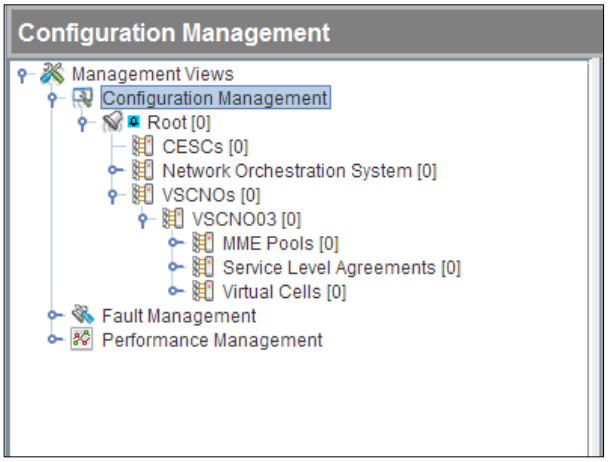

Figure 13 – VSCNO03 Configuration Management view

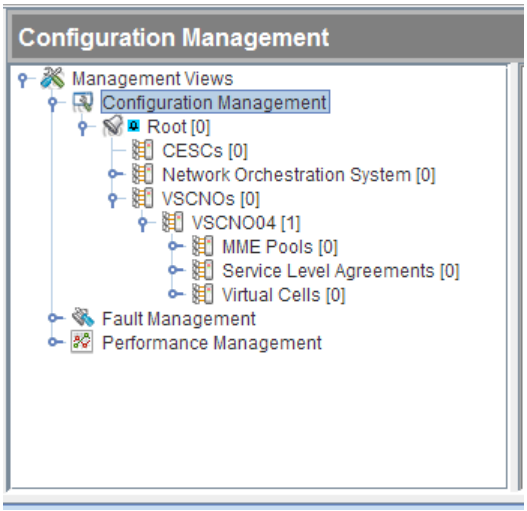

Figure 14 – VSCNO04 Configuration Management view

3.1.6 Northbound Interfaces

3.1.6.1 Configuration Management

The EMS supports a special class of northbound user that is allowed to perform configuration management operations via a SOAP[8] interface that is compliant with the 3GPP Configuration Management IRPs ([11], [12], [13] and [14]).
These users may be enrolled with equivalent permissions to those of the EMS client GUI in order to provide network isolation.

Although fully implemented, to date extensive testing of this interface has not been performed.

3.1.6.2 Performance Management

3.1.6.2.1 VSCNO Specific PM Reports

On receipt of Performance Management XML files from the PNF, the EMS parses these into separate copies for each VSNO hosted on the same CESC and the PNF.
Each PM counter falls into one of three categories that control how it is processed and what appears in the output per-VSCNO version:

- Common counters are copied, unmodified, into each output file. For example, a counter that is considered of common interest is *Cell Unavailable Time* which is a standard 3GPP counter defined in [15].

- SCNO private counters are not copied into individual per-VSCNO output files as they may disclose information that the SCNO does not want to reveal. Every counter for which there is an associated per-VSCNO version falls into this category as it is possible that, given such a value, a VSCNO could derive the value of another VSCNO's counters by subtracting their own values from the PNF-wide value.

[8] For further details, also see, *among others*: https://en.wikipedia.org/wiki/SOAP.

- Filtered versions of the VSCNO specific counters detained in section 3.1.2 are copied into individual per-VSCNO output files such that each file contains only the values that are relevant to the VSCNO concerned; i.e., the ones that contain the VSCNO's PLMN.

Once created, these VSCNO-specific PM files are available in a northbound file structure dedicated to the VSCNO.

3.1.6.3 VSCNO Specific KPI Reports

Once the EMS has received and processed a Performance Management XML file, the calculated KPI values are also made available as an XML file in a VSCNO specific northbound directory.

Note that the values on these files are computed solely upon the time frame covered by the corresponding PM XML file (its *granularity period*) and are not related to the temporal scope of Monitored SLAs which may aggregate values from multiple PM reports.

3.1.6.4 Fault Management

Due to effort constraints, the northbound fault management interface described in Deliverable D2.4 [16] (3GPP 32.111 [17]) has not been implemented.

3.2 Dynamic Resource Allocation

In the SESAME environment, resource allocation is the process of appointing available resources dynamically to the VNFs and the applications.
More in general, in a cloud environment the resource allocation is based on the infrastructure as-a-service (IaaS) model, which also consists one of SESAME's main functioning model. Resource allocation techniques should be optimized to avoid resource contention, resource fragmentation and over provisioning of resources. There might be a situation where two applications try to access the same resource at the same time, while another case may appear where there are limited resources and the demand for resources is high.
Resource allocation techniques should satisfy multiple applications which need different types of resources such as CPU and memory.

Typically, Virtual Machine monitoring systems provide a mechanism for mapping virtual machines (VMs) to physical resources. This mapping needs to be largely hidden from the end-users. VM live migration technology makes it possible to change the mapping between VMs and physical machines (PMs) while applications are running.
However, a policy issue remains as how to decide the mapping adaptively so that the resource demands of VMs are met, whilst the number of PMs used is minimized. This is challenging when the resource needs of VMs are heterogeneous due to the diverse set of applications they run and vary with time as the workloads grow and shrink. The capacity of PMs can also be heterogeneous because multiple generations of hardware may co-exist in a data centre.

Since node resources are shared, providing guarantees to applications in the shared data centre model is more complex. Typically, such guarantees are provided by reserving a certain fraction of node resources (CPU, network and disk) for each application.
The fraction of the resources allocated to each application depends on the expected workload and the QoS requirements of the application. The workload of web applications is known to vary dynamically over multiple time scales and it is challenging to estimate such workloads *a priori*.

Consequently, static allocation of resources to applications is problematic, while over-provisioning resources based on worst case workload estimates can result in potential underutilization of resources, under provisioning resources can result in violation of guarantees.
An alternate approach is to allocate resources to applications dynamically based on the variations in their workloads. In this approach, each application is given a certain minimum share based on coarse-grain estimates of its resource needs; the remaining server capacity is dynamically shared among various applications based on their instantaneous needs.

3.3 Large Scale Deployment

The fundamental concept of SESAME is the virtualization of the small cell and its partitioning to logically isolated slices, followed by an integration with a virtualized execution infrastructure.

The Proof of Concept demos are described below:

- Multi-tenancy and Monitoring: The SESAME PoC illustrates the establishment of the complete chain of monitoring, decision-making and reaction. In this case, the CESCM as a module with the overall view of the radio and cloud side of the ecosystem monitors cloud/radio parameters (e.g., CPU/RAM usage, call drop rate, etc.) If a violation occurs, the CESCM, via processing of the monitoring inputs, will be able to detect and then appropriately react to the situation. The decision-making process might turn out to be a simple threshold checking or a complicated multi-parameter cognitive method. In the same way, the reaction ranges from the complete network service (NS) scaling, to the NS scaling up/down, in/out, to the service function chain changes, to the change on a radio parameter (e.g., dedicated bandwidth to a VSCNO).

- Service Chaining: The Light DC is the component where all VNFs, SC-VNFs and resulting SFCs are executed. It provides heterogeneous platform consisting of ARMv8[9] and/or x86[10] nodes. Some of them can be equipped with different hardware accelerators (such as FPGA, GPU) enabling offloading of heavy computational tasks (e.g., video transcoding, etc.) from the CPU. This hardware is fully supported by the software baseline providing virtualization, virtualized hardware accelerators, accelerated virtual networking as well as integration with the SESAME Virtual Infrastructure Management (VIM) of choice, i.e., OpenStack[11].

[9] For more related information, see: *http://www.arm.com/products/processors/armv8-architecture.php*

[10] The *x86* is a family of backward compatible instruction set architectures based on the Intel 8086 CPU and its Intel 8088 variant. More related information can be found, for example, at: *https://en.wikipedia.org/wiki/X86*.

[11] For more related information see: *https://www.openstack.org/*.

4 Discussion

This section describes the components that deliver the use cases and assesses how well the Proof of Concept demonstration demonstrates them.

4.1 MOCN

The ip.access E40 [4]Access Point used to provide the SESAME SC PNF functionality supports all of the RAN features and procedures of standard MOCN.
However, its implementation of the core network facing features is restricted to a single S1 connection in expectation that there is a gateway (GW), providing a fan-out function, between it and the core network (CN) of each connected operator.
This expectation aligns perfectly with the SESAME PoC architecture where the VNFs hosted by the CESC provide the required fan-out and multiplexing functionality.

Early testing validated that:

- The ip.access E40 was able establish an S1 connection to the EPC provided by Athonet.
- That a single S1 connection could be used to support multiple PLMN IDs[12].
- That the E40 broadcast the required PLMN IDs and that different UEs with different home networks would select the appropriate PLMN, which was communicated via S1 to the EPC.

Subsequent testing with the full PoC architecture validated that:

- The SC PNF (ip.access E40) was able establish an S1 connection to the SC Common VNF which provides the S1 multiplexing and fan-out functionality within the CESC.
- Each SC VNF is able to establish a separate S1 connection to the EPC (or indeed any EPC).
- The S1 connection established by each SC VNF supports only the PLMN ID of the Virtual Network Operator served by the SC VNF.
- UEs with different home networks have their S1 and GTP traffic routed via the appropriate SC VNF towards the specified EPC.

4.2 Network Isolation

Network Protocol traces taken on an emulated[13] CESC platform have been used to validate that S1 and GTP traffic is routed correctly.
For example:

- That an S1 INITIAL UE message for a specific PLMN is routed first to the SC Common VNF and then to the correct SC VNF and onward to the serving EPC.

[12] Although not standard MOCN, the 3GPP specifications allow for this “Gateway Core Network” configuration.
[13] Effort has not permitted the SC-Common VNF and SC VNF to be ported to the ARM platform. This has, thus, been emulated on a PC platform running Oracle VirtualBox [19].

- Importantly, that this message **is not** copied to other SC VNFs and does not find its way onto the backhaul of the wrong EPC.

- That user plane traffic for a specific VSCNO is routed via the correct SC VNF and bypasses the SC Common VNF, which is only responsible for S1 signalling.

- That the user plane traffic of one SC VNF is not received by other SC VNFs and, therefore, does not place any load on them.

As reported in Deliverable D7.3, measurements taken with the Linux "top" command during a set of throughput tests have been used to validate that:

- The load placed on the SC Common VNF was, as expected, minimal. It spiked briefly at call set-up time and was then effectively zero for the duration of the call with another brief spike at call tear-down.

- The load placed on the SC VNF was, as expected, dominated by the user plane throughput and directly proportional to it,

- There was no data leakage leading to load on the other, idle, VNFs running on the same CESC.

4.3 Per-VSCNO Network Slice

4.3.1 Provisioned SLA

A VSCNO's network slice is defined by a special class of managed object that specifies:

- The maximum number of UEs belonging to VSCNO's PLMN that may be active on the cell at any one time.

- The maximum, aggregate, uplink throughput that a VSCNO's UEs may generate.

- The maximum, aggregate, downlink throughput that a VSCNO's UEs may generate.

At virtual cell provisioning time, the values specified in the Provisioned SLA are propagated to the configuration of the SC VNF serving the VSCNO and are used by it to police the load that can placed on it my UEs belonging to the VSCNO's PLMN.

4.3.2 UE Admission Cap

As reported in Deliverable D7.3, the per-VSCNO UE limit appears to work as expected and can be changed on-the-fly, as required.

4.3.3 Per VSNO Throughput Cap

4.3.3.1 Initial Algorithm

The algorithm selected initially was very simple. Remember that each SC VNF only handles the user plane traffic of one VSCN. It is configured with separate uplink and downlink throughput by means of the Provisioned SLA object used during the Virtual Cell creation process (see 3.1.1.11).

It then operates a one-second tick:

- At the beginning of each second, the SC VNF resets its two cumulative bit counts for uplink and downlink to zero.

- On receipt of a GTP packet, it extracts the payload length (see **Figure** 18), which is measured in bytes, multiplies this value by 8 to yield the payload bit count and then adds the resulting value to the accumulator for the associated direction (uplink or downlink).

- If the value of a bit count accumulator exceeds the configure value, the SC VNF enters discard mode. All subsequent GTP packets received for the associated direction are discarded until the end of the second when the accumulator value is reset to zero.

4.3.3.2 Initial Results

Early testing of the throughput capping mechanism with a publically available throughput test [5] showed that, whilst the measured throughput was proportional to the configured limit, the delivered rate was roughly half that of the configured value as illustrated by Table 2 below:

Configured UL Throughput (MBits/s)	2	5	10	15	20
Configured DL Throughput (MBits/s)	2	5	10	15	20
Measured UL Throughput (MBits/s)	0.912	2.844	6.112	9.486	16.1
Measured DL Throughput (MBits/s)	0.898	2.104	4.268	7.916	9.374
Measured UL Throughput Percent	45.6%	56.9%	61.1%	63.2%	80.5%
Measured DL Throughput Percent	44.9%	42.1%	42.7%	52.7%	46.9%

Table 2 – Initial Throughput Cap Test Results

Analysis of the input and output data streams revealed that, although the throughput throttling algorithm was correctly discarding GTP-U packets during peaks of activity, it was making no allowance for periods of inactivity. Figure 15 and Figure 16 below illustrate the nature of the input data stream as captured by Wireshark [18]. Figure 16 has been annotated to illustrate that, if the throughput limit was set at a level indicated by the red line, the data in the peaks shaded in orange would be discarded. However, no allowance is made for the areas shaded in light blue where throughput does not reach the configured limit.

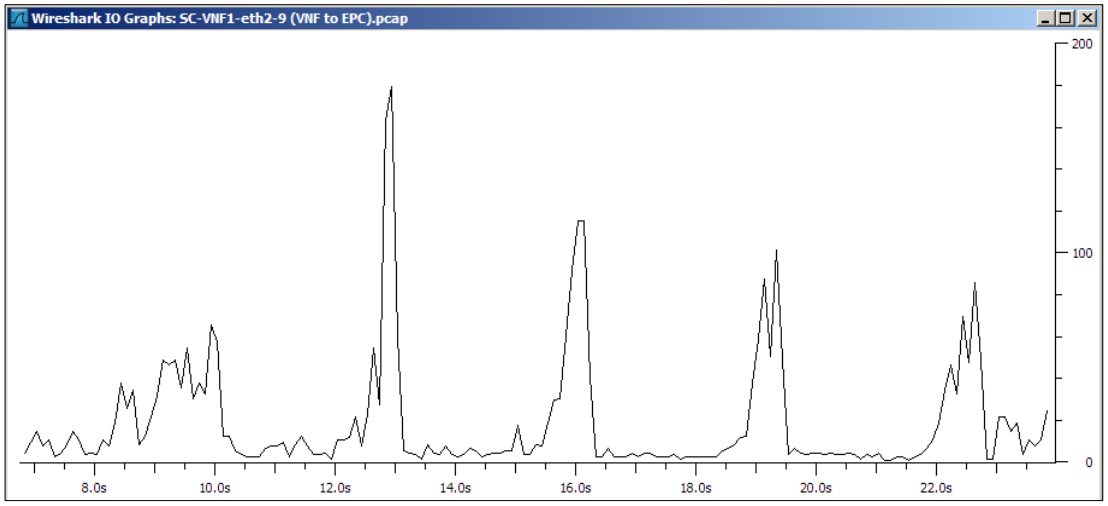

Figure 15 – Bursty Input Data Stream – Packets per second

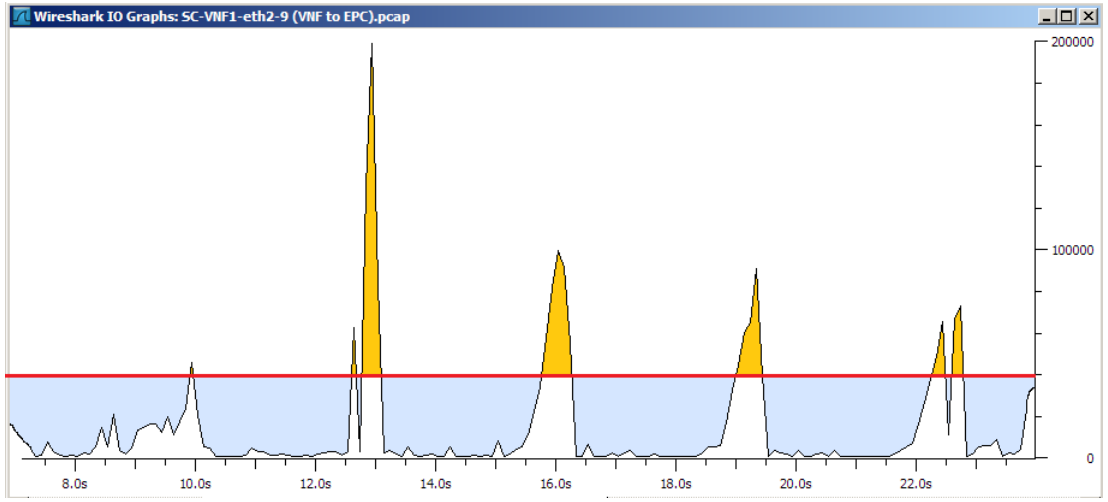

Figure 16 – Bursty Input Data Stream – Bytes per second

4.3.3.3 Amended Algorithm

The throttling algorithm was amended to introduce a "credit" concept. During periods of inactivity, the VNF accumulates throughput credit, in each direction separately, up to a limit of N times the configured maximum throughput for each direction (UL or DL).
It is then able to burst at a peak rate up to this limit until all the credit is consumed. Thereafter, the configured limit is applied until new credit is accumulated following further periods of inactivity.
Thus, short term peaks of activity may be accommodated whilst maintaining the overall throughput limit.

4.3.3.4 Amended Algorithm Results

Setting the value of N to 10 yielded the following test results:

Configured UL Throughput (MBits/s)	2	5	10	15	20
Configured DL Throughput (MBits/s)	2	5	10	15	20

Measured UL Throughput (MBits/s)	1.998	4.48	7.90	12.88	13.77
Measured DL Throughput (MBits/s)	1.698	3.77	8.39	11.44	12.25
Measured UL Throughput Percent	100%	90%	79%	86%	69%
Measured DL Throughput Percent	85%	75%	84%	76%	61%

Table 3 – Final Throughput Cap Test Results

Whilst it would certainly be possible to improve the algorithm further, to some degree it would be optimised to conform to the behaviour of the particular test being performed [19]. These results are considered adequate for the purposes of the PoC demo.

4.3.4 Service Chaining

Service chaining is provided as a configurable option by the SC VNF. When enabled, it causes the VNF to present user plane traffic on a pair of dedicated Ethernet interfaces for routing into the service chain. The uplink and downlink traffic streams may be configured separately (see Figure 17 below), allowing a high degree of flexibility.

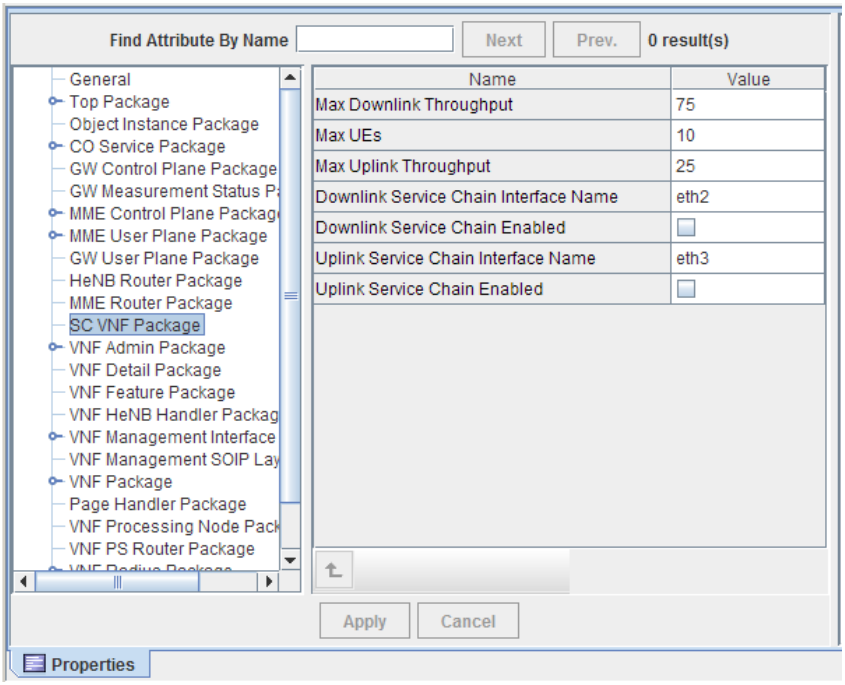

Figure 17 – VNF Service Chain Configuration

When enabled, the SC VNF extracts the payload from each GTP packet, indicated by the yellow elements of **Figure** 18 below, and forwards it on the configured interface.

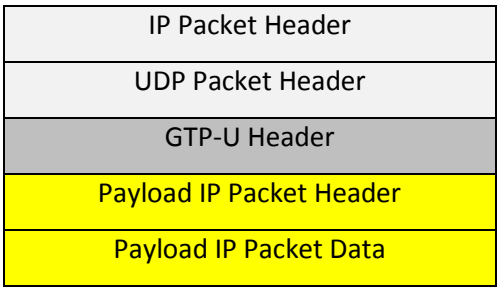

Figure 18 – GTP Protocol Stack

On receipt of an IP packet on this interface, the VNF identifies the original GTP tunnel, encapsulates it with GTP and forwards it in the appropriate direction.

The details of this mechanism are described in *Appendix A*.

4.4 CESCM and EMS – SLA Integration

4.4.1 EMS Functionality

On receipt of a Performance Management XML file from a PNF, the EMS performs a number of steps that include:

- Parsing the file to discover the identity of the PNF as defined by its Equipment Identity (EID), present in the name of the file.
- Extract the time-frame covered by the file.
- Extract the various physical cell and virtual cell counter values.
- Compute the associated KPIs and store them in the EMS database for subsequent use.

These stored KPI values have been made available to the CESCM by means of two elements; a shell script, that runs once per minutes as a CRON job [9], and which interrogates the EMS database for updated KPI values and a Web server that publishes the extracted data in JSON format [10] on a specific IP address and port number combination for retrieval by northbound systems such as the CESCM.

An example of the published data for two PNFs is presented in Figure 19 below:


```
[
{
    "help": "Availability of the PNF measured as a percentage",
    "metric_type": "gauge",
    "collect": [
        {"metric_name": "sc_availability", "label": {"sc": "physical", "eid":
             "000295-0000272279", "value": "99.916666666"}}
        {"metric_name": "sc_availability", "label": {"sc": "physical", "eid":
             "000295-3141592714", "value": "99.916666666"}}
    ]
},
{
    "help": "The total number of uplink octets transmitted by the PNF for the
             virtual cell",
    "metric_type": "gauge",
    "collect": [
        {"metric_name": "uplinkOctetsKpiName", "label": {"sc": "virtual", "eid":
             "000295-0000272279", "value": "2000"}}
        {"metric_name": "uplinkOctetsKpiName", "label": {"sc": "virtual", "eid":
             "000295-3141592714", "value": "2000"}}
    ]
},
{
    "help": "The total number of downlink octets received by the PNF for the
             virtual cell",
    "metric_type": "gauge",
    "collect": [
        {"metric_name": "downlinkOctetsKpiName", "label": {"sc": "virtual",
             "eid": "000295-0000272279", "value": "1000"}}
        {"metric_name": "downlinkOctetsKpiName", "label": {"sc": "virtual",
             "eid": "000295-3141592714", "value": "1000"}}
    ]
},
{
    "help": "Call drop rate of the PNF measured as a percentage",
```

```
    "metric_type": "gauge",
    "collect": [
    ]
}
]
```

Figure 19 – Example JSON KPI Data

Retrieved values are published by the Web server in response to an HTTP GET and are retained until superseded by more recent values.
Thus, an HTTP GET on this IP address and port number combination always retrieves the most up-to-date values.
For a single given PNF, the values are typically updated once per hour as new PM XML files are received by the EMS. However, on a system with multiple PNFs, uploading data in a distributed fashion, the values reported on this interface will change frequently.

4.4.2 Dashboard Functionality

As described in Deliverable D5.2 [20], Prometheus was selected as third party monitoring tool to assess the contract terms.
Prometheus [21] is an open source white box monitoring and alerting system that is designed for large and scalable environments. It provides a flexible query language, allowing slicing and dicing of collected time-series data in order to generate ad-hoc graphs, tables, and alerts and is integrated with visualization tools (Grafana[14] and PromDash[15]).
The data is collected in a "pull" based model, developed as exporters; allowing bridging the internal state of the application.

Depending on the type of value, metrics can be defined by one of the following types:

- A counter is a metric which is a numerical value that is only incremented, never decremented. Examples include the total amount of requests served, how many exceptions that occur, etc.

- A gauge is an instantaneous metric value that is created via incrementing, decrementing or accumulation. An example could be memory usage, CPU usage, amount of threads, or perhaps a temperature.

- A histogram is a metric that samples observations. These observations are counted and placed into configurable buckets. Upon being scraped, a histogram provides multiple time series, including one for each bucket, one for the sum of all values, and one for the count of the events that have been observed. A typical use case for a histogram is the measuring of response times.

- A summary is similar to a histogram, but it also calculates configurable quantiles. Depending on your requirements, you either use a histogram or a summary.

A data exporter for SESAME EMS, as described in section 4.4.1 above, uses the gauge metric exclusively to retrieve and present KPI values.

[14] For further details also see: *https://grafana.com/*.
[15] For further information also consider, *inter-alia*: *https://github.com/Accenture/tldr-promdash*.

5 Conclusion

This section discusses how well the PoC Demonstration achieved the original objectives and highlights any areas for future research or development.

5.1 Demo Architecture

5.1.1 MOCN

The PoC demo successfully demonstrates full MOCN functionality across the intended architecture, which is designed to support the requirements of network isolation and per-VSCNO network slices described below. As discussed in Deliverable D7.5 [22], areas where this functionality might be improved include:

- A better way of marking a virtual cell as unavailable instead of the current mechanism of setting the *Cell Reserved for Operator Use* element broadcast in System Information Block 1 (SIB1) by the PNF. This would, however, require a change to the 3GPP standards that would have be achieved in a backwards compatible manner.

- Additional per-PLMN / per-VSCNO performance counters; especially in the area of handover.

- Per VSCNO paging caps. Paging requests are another area where a badly behaved VSCNO could compromise the performance of the cell for other VSCNOs. Currently, the PNF discards paging requests (for all VSCNOs) when its buffers reach capacity. By implementing a per-VSNO cap, only the badly behaving VSCNO would have their paging requests discarded and other VSCNOs hosted on the same VNF would continue to operate normally.

5.1.2 Network Isolation

As described in section 4.2, the PoC demo successfully demonstrates the key aspects of network isolation including:

- The processing of user plane traffic by a separate SC VNF per VSCNO. Each SC VNF runs in a dedicated virtual machine and is, thus, largely unable to influence the other SC VNFs running on the same CESC.

- The separation of both signalling user plane traffic through each SC VNF so that, if required, this traffic can be carried on a separate backhaul into the appropriate EPC.

- The separation of service chains for each VSCNO. When enabled by configuration, each VSCNO's user plane traffic may be routed through a different service chain as required.

As intended in the original design, the PoC architecture allows the computing resources assigned to a VNF (i.e. CPU and to a lesser extent RAM) to be scaled according to the anticipated load and to subsequently be adjusted in response to live performance measurements.

5.1.3 Per-VSCNO Network Slice

As described in sections 4.3.2 and 4.3.3, the PoC demo successfully demonstrates the feasibility of capping the maximum number of UEs on a per-VSNO basis and of applying separate limits to the total uplink and

downlink bandwidth that each VSCNO may consume. Although the bandwidth throttling algorithm is fairly simple and has been tuned to match the particular test, the results demonstrate that the concept is sound and can be achieved without placing a high degree of additional load on the SC VNF.

Possible areas for future research include:

- Alternative throttling / traffic shaping algorithms;
- Capping paging requests on a per-VSCNO basis, as discussed in section 5.1.1;
- More sophisticated and flexible capping schemes. Currently, the PoC imposed a hard limit on both the maximum number of active UEs and the maximum uplink and downlink throughput. A more sophisticated scheme might allow for a set of limits such as:
    - A minimum guaranteed level of resource that a VSCNO can always expect to attain;
    - A maximum resource level that the VSCNO can never exceed;
    - The ability to consume resources between these two limits when they are not in use by another VSCNO on the same CESC.

5.1.4 CESCM – EMS SLA Integration

As described in section 4.4, the PoC demo provides an interface, based on HTTP, between the EMS and the CESCM (the CESCM Portal in Figure 1) for the reporting of SLA information by the EMS. Currently, this information includes:

- The availability of the PNF as a percentage during the reporting period;
- The uplink and downlink octets transmitted and received by the PNF;
- The uplink and downlink packets transmitted and received by the PNF.

Whilst it successfully demonstrates the concepts of an SLA reporting interface and the ability, it is currently very basic in nature:

- For the purposes of the PoC demonstration, the number KPIs reported is limited to just the five listed above;
- Data is updated as it arrives and is processed by the EMS. It takes no special action on late or missing data in the event that a PNF has been reset or powered off;
- There is no security on the interface which exchanges data in plain text, using HTTP.

Despite these limitations, the EMS – CESC interface successfully demonstrates the ability to integrate management of the PNF into the dashboard provided by CESC Manager, to retrieve multiple KPI values and display them in a graphical manner.

5.2 Use Case Scenarios

The following use cases were evaluated, using the PoC demonstration.

5.2.1 Multi-Tenancy

Multi-tenancy is perhaps the most fundamental of SESAME concepts. It is delivered by a combination of four main elements, as discussed below:

- Standard 3GPP MOCN functionality, which provides RAN sharing and the ability for UEs with different home PLMNs to access a shared cell and have their traffic routed to the appropriate core network.

- Per-VSNO resource limits. This is a "key" extension to standard MOCN functionality that changes resource management from a first-*come, first-served* basis to a managed framework where not only does each VSCNO obtain their fair share of resources, but this share is defined by an agreed SLA which can be different for each participating VSCNO.

- Network isolation provided by the VNFs running on the CESC. This not only isolates the traffic of each VSNO from each other VSCNO but also the load that this traffic imposes.

- Network isolation provided by the EMS which isolates the management views of each VSCNO from each other and provides them with the ability to manage their virtual network in a manner analogous to a physical network with minimal interaction required by the host SCNO.

The PoC demonstration successfully demonstrates all of the above multi-tenancy aspects.
Possible areas for improvement include:

- As discussed in Deliverable D7.5 [22], improving information about the operability of virtual cells based upon the status of the PNF and SC VNF components that host it.

- Improving network isolation on the EMS northbound interfaces so that VSCNOs can use these as a means to integrate into their own systems.

- Providing an option to re-home a virtual cell on to a different CESC. Although not fully implemented by the current PoC, the conceptual virtual cell provisioning process is as follows:

    - The VSCNO specifies both a desired location for service and a desired SLA for the new virtual cell.

    - The EMS offers the VSCNO a list of cells, in order of proximity to their desired location, that have sufficient spare capacity to deliver the requested SLA.

    - The VSCNO selects a CESC from this list and the virtual cell is provisioned on it.

    Should a CESC be decommissioned or a better match to the VSCNO's requirements arise, there is currently no means of re-homing a virtual cell onto a different CESC. This would be a highly desirable feature in a commercial deployment.

5.2.2 Dynamic Resource Allocation

Whilst not fully implemented, the SESAME PoC demonstrates that the fundamental building blocks for dynamical resource allocation are present and working:

- The CESCM is able to monitor metrics reporting on the performance of both the PNF and each VSCNO's VNF.

- A VSNO's network slice, in terms of maximum number of UEs and maximum uplink and downlink throughput, can be modified on-the-fly by means of the EMS' northbound interface.

Combining these aspects to dynamically alter resource allocation in response to changing conditions, *as they occur*, is a small additional step that would enable:

- Scaling up and down of resources in response to load.
- VSCNOs to have more sophisticated SLAs with, for example, minimum and maximum desired resource levels. Thus, for example, a VSCNO's virtual cell might be provisioned on a CESC that supports their minimum required SLA but subsequent changes, such as the departure of another VSCNO, might enable their maximum required SLA to be supported.

5.2.3 Large Scale Deployment

As discussed in section 5.2.1 above, the PoC successfully demonstrates the multi-tenancy aspects required for a large scale deployment and the path from the current PoC to a commercial deployment is clear.

Further work is required in the area of service chaining.
The PoC implements a service chain interface that has been tested in a stand-alone manner and, to date, this has not be integrated with the demonstration service chain VNFs.

6 Appendix A – SC VNF Service Chaining Implementation

The SC VNF provides the option to de-encapsulate GTP packets and inject the payload IP packet into the service chain. How the packet is processed by the VNFs that form the service chain is not the responsibility of the SC VNF but in general; there are a small number of possible outcomes:

- No Service Chain VNF is able to process the packet and it is returned, unmodified, to the SC VNF.
- A Service Chain VNF, such as vDPI (see section 2.3.2), decides to drop the packet.
- One or more Service Chain VNF, such as vWatermark, modify the packet or packet stream and return modified data to the SC VNF.
- A Service Chain VNF acting as a proxy for some entity normally reached via the EPC intercepts the packet and generates a response.

Given that the SC VNF will be handling at least one GTP tunnel per active UE, for any packet received on the Service Chain Interface, it must decide both which GTP tunnel it belongs to and in which direction (uplink or downlink) it must be sent[16],[17].
It does this as follows:

- On receipt of a GTP packet, the VNF inspects the IP packet header of the payload and uses the combination of source and destination IP address and IP port number to identify a "data flow" associated with a given GTP Tunnel Endpoint Identifier (GTP TEID);
- For packets in the reverse direction the IP address and port numbers will be, *similarly*, reversed and there will be a separate GTP TEID. When it identifies a reverse flow, the SC VNF correlates the uplink and downlink TEIDs into a pair associated with the flow.
- When service chaining is enabled, the SC VNF de-encapsulates received GTP packets, inspects the IP packet header of the payload, as described above, prior to forwarding it on the configured interface.
- On receipt of an IP packet on a service chain interface, the SC VNF inspects the IP packet header and compares the source and destination IP address and port number with those of identified data flows. If a match is found, the packet is encapsulated with a GTP header and forwarded in the appropriate direction. If no match is found, the packet is discarded,

[16] For example, a proxy VNF handling an uplink packet might generate multiple downlink packets in response.
[17] Note that for a given active RAB there are separate uplink and downlink GTP tunnels.

7 References

[1] 3rd Generation Partnership Project (2012): *3GPP TS 36.300, E-UTRAN Overall description; Stage 2*. 3GPP.

[2] 3rd Generation Partnership Project (2012): 3GPP TS 23.236, Intra-domain connection of Radio Access Network (RAN) nodes to multiple Core Network (CN) nodes. 3GPP.

[3] 5G-PPP SESAME Project (2016): *Deliverable 2.5: SESAME Final Architecture and PoC Assessment KPIs.*

[4] ip.access Ltd. (2017): E*40 nanoLTE, 4G Enterprise Access Point Data Sheet.* Available at: ht*tp://www.ipaccess.com/uploads/wysiwyg_editor/files/2017/E40-Datasheet-v1.0.pdf*

[5] Ookla (2017): *Speedtest*. [Online]. Available at: http://www.speedtest.net/.

[6] 5G-PPP SESAME Project (2017): *Deliverable 7.2: Integrated CESC Prototype Validation*.

[7] ip.access Ltd (2017): *NANO_GST_14005, NOS Product Description*.

[8] 3rd Generation Partnership Project (2012): *3GPP TS 32.435: Telecommunication management; Performance measurement; eXtensible Markup Language (XML) file format definition (Release 10)*. 3GPP.

[9] Wikipedia: *Cron*. [Online]. Available at: *https://en.wikipedia.org/wiki/Cron.*

[10] ECMA (2013, October): *Introducing JSON*. [Online]. Available at: *http://www.json.org*.

[11] 3rd Generation Partnership Project (2010, September): *3GPP TS 32.306, Configuration Management (CM); Notification Integration Reference Point (IRP): Solution Set (SS) definitions (Release 10)*. 3GPP.

[12] 3rd Generation Partnership Project (2010, June): *3GPP TS 32.316, Generic Integration Reference Point (IRP) management; Solution Set (SS) definitions (Release 10)*. 3GPP.

[13] 3rd Generation Partnership Project (2010, September): *3GPP TS 32.606, Configuration Management (CM); Basic CM Integration Reference Point (IRP); Solution Set (SS) definitions (Release 10)*. 3GPP.

[14] 3rd Generation Partnership Project (2011, March): *3GPP TS 32.662, Configuration Management (CM); Kernel CM Information Service (IS) (Release 10)*. 3GPP.

[15] 3rd Generation Partnership Project (2012): *Telecommunication management; Performance Management (PM); Performance measurements, Evolved Universal Terrestrial Radio Access Network (E-UTRAN)*. 3GPP.

[16] 5G-PPP SESAME Project (2015, December): *Deliverable 2.4: Specification of the Infrastructure Virtualisation, Orchestration and Management – First Iteration*.

[17] 3rd Generation Partnership Project (2011): *3GPP TS 32.111, Telecommunication management; Fault Management; Part 2: Alarm Integration Reference Point (IRP): (Release*

10). 3GPP.

[18] Wireshark [Online]. Available at: *https://www.wireshark.org/*.

[19] Ookla: *Speedttest* [Online]. Available at: *http://www.speedtest.net/*.

[20] 5G-PPP SESAME Project (2017, September): *Deliverable 5.2: VIM and CESCM Implementation*.

[21] Open Source Community: Prometheus - From metrics to insight [Online]. Available at: *https://prometheus.io/*.

[22] 5G-PPP SESAME Project (2017): *Deliverable 7.5: Overall Assessment and Roadmap*.

[23] ITU (1992): *CCITT Recommendation*: *X.731 Information technology – Open Systems interconnection – Systems Management: State Management Function*. International telecommunication Union (ITU).

[24] Broadband Forum (BF) (2013, November); *TR-069 CPE WAN Management Protocol, Issue:1, Amendment 5.*

[25] ORACLE CORPORATION (2017): *Virtual Box 5.1*. [Online]. Available at: *https://www.virtualbox.org/*.

.