

Open Source
MANO

Demonstrating 5G Core network
automation by OSM

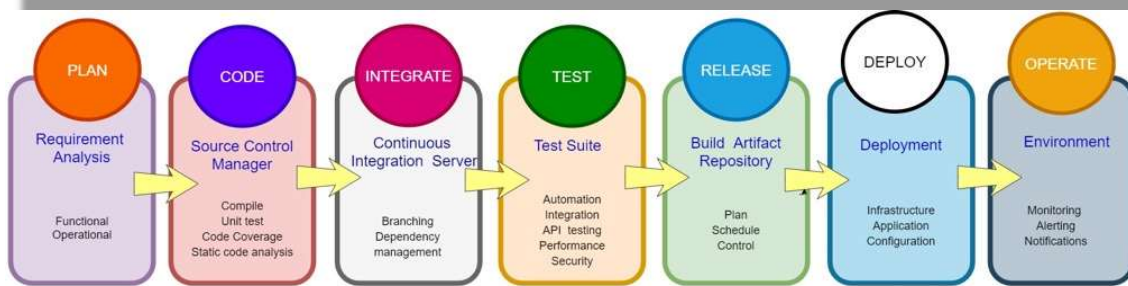
ULAK Communications

Gülsüm Atıcı & Naciye Akyıldız

ÇINAR 5G Core



- Cross-functional development teams from ULAK and partners are co-developing 5G Core network (Çinar) according to 3GPP standards, utilizing agile development methodology (Scrum) and using smart, continuous integration and continuous delivery pipeline.



- Source Management Repository & Release management: **Bitbucket**
- Static code check: **CppCheck**
- Test Automation Tools: **NodeJs, Npm, Mocha**
- Build Automation Tool: **GNU Make**
- Continuous Integration Tool : **Jenkins**
- Code Coverage Tool: **Gcovr**

Cppcheck Results

Summary

Severity	Count	Delta
Error	0	
Warning	0	
Style	212	
Performance	0	
Portability	0	
Information	2	
No category	0	
Total	214	

Details

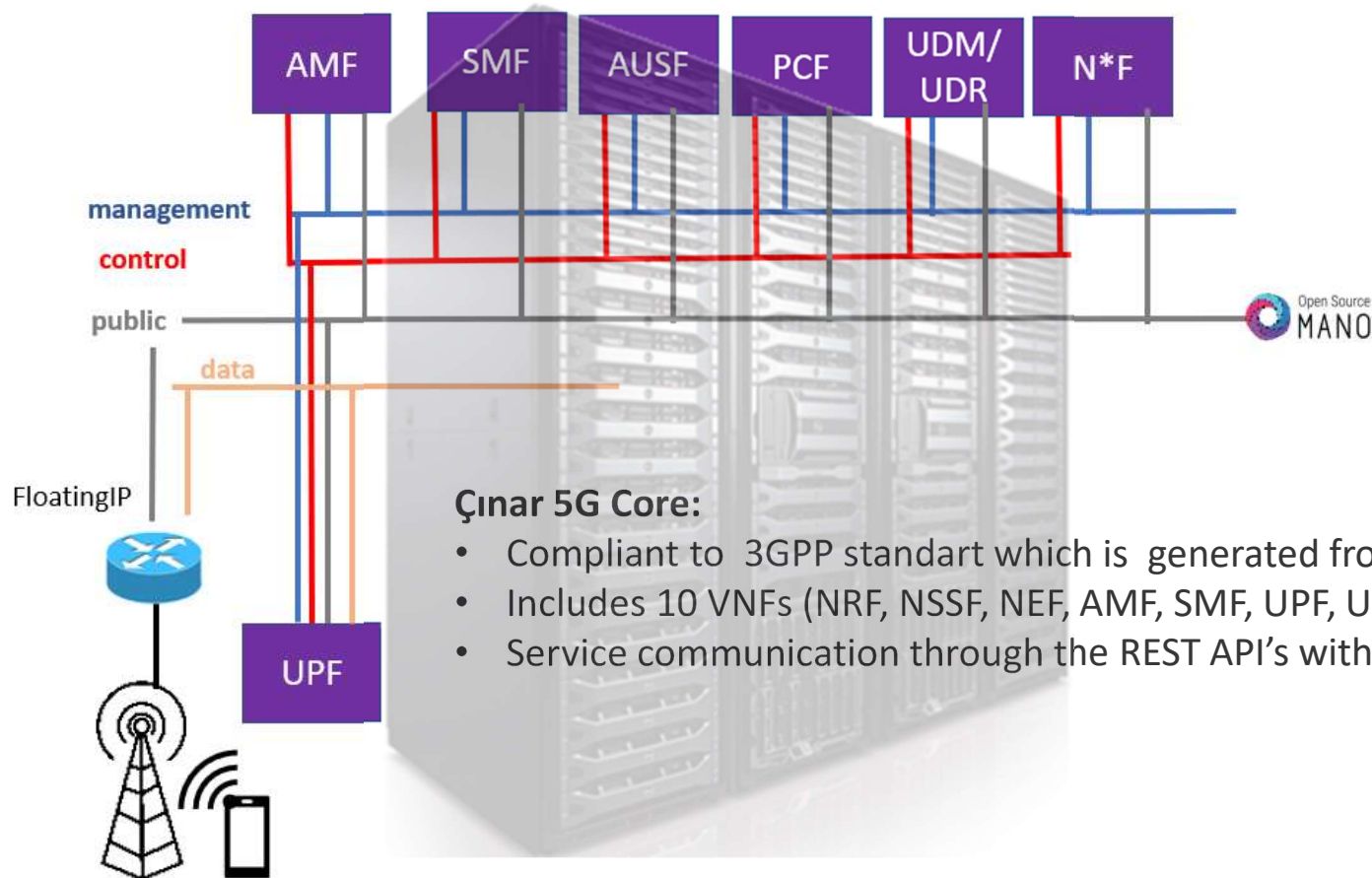
Show issues highlighted on a single page

- all
- error,warning
- style
- information

State	File	Line	Severity	Type	Inconclusive	Message
unchanged	src/systemServices/Communication/IndividualSubscriptionDocumentServer.h	17	style	reduplicatedConstructor	false	Class 'IndividualSubscriptionDocumentServer' has a constructor with 1 argument that is not explicit.
unchanged	src/systemServices/Communication/IndividualSubscriptionDocumentServer.cpp	113	style	unusedVariable	false	Variable 'contentId' is assigned a value that is never used.
unchanged	src/systemServices/Communication/IndividualSubscriptionDocumentServer.h	18	style	reduplicatedConstructor	false	Class 'IndividualSubscriptionDocumentServer' has a constructor with 1 argument that is not explicit.
unchanged	src/systemServices/Communication/IndividualSubscriptionDocumentServer.h	17	style	reduplicatedConstructor	false	Class 'IndividualSubscriptionDocumentServer' has a constructor with 1 argument that is not explicit.



ÇINAR 5G Core

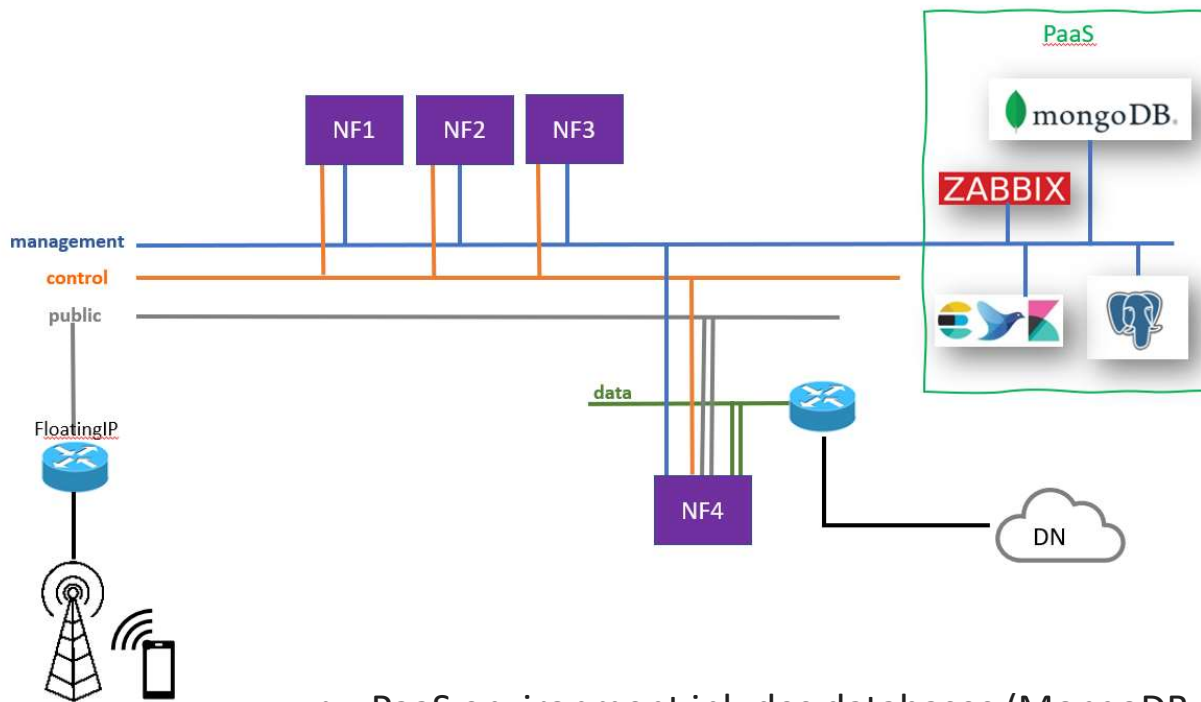


Çınar 5G Core:

- Compliant to 3GPP standard which is generated from 3GPP 5G YAML
- Includes 10 VNFs (NRF, NSSF, NEF, AMF, SMF, UPF, UDM, UDR, AUSF, PCF)
- Service communication through the REST API's within server /client model

- ULAK VIM solution is used for development, test and deployment
- We are extending it to participate OSM events as VIM/NFVi /MEC provider

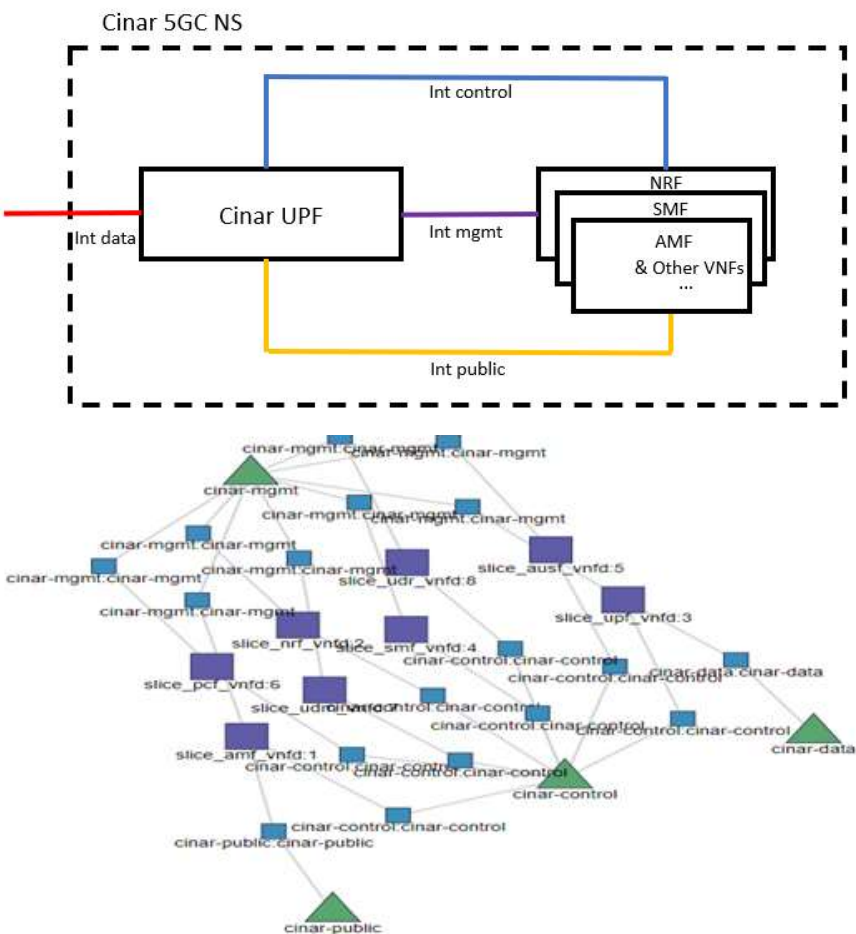
ÇINAR 5G Core



- Utilizes 4 separate networks
- Management network for connection to database and other tools
- Control network is used for communication between NF's
- Public network is used for external access
- Data network is used for data transportation

- PaaS environment includes databases (MongoDB & Postgresql)
- Monitoring (Zabbix), log collection and analytics tools (EFK)

ÇINAR 5GC NSD Design



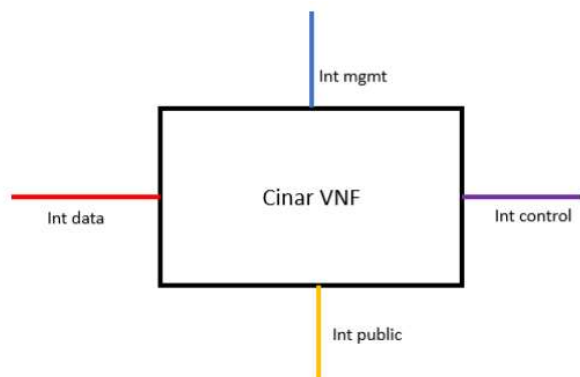
```
constituent-vnfd:
- member-vnf-index: 1
  vnfd-id-ref: mslice1_nrf_vnfd
- member-vnf-index: 2
  vnfd-id-ref: mslice1_pcf_vnfd
- member-vnf-index: 3
  vnfd-id-ref: mslice1_smf_vnfd
- member-vnf-index: 4
  vnfd-id-ref: mslice1_ausf_vnfd
- member-vnf-index: 5
  vnfd-id-ref: mslice1_udr_vnfd
- member-vnf-index: 6
  vnfd-id-ref: mslice1_udm_vnfd
- member-vnf-index: 7
  vnfd-id-ref: mslice1_amf_vnfd
- member-vnf-index: 8
  vnfd-id-ref: mslice1_upf_vnfd
```

```
connection-point:
- name: cinar-mgmt
  vld-id-ref: s1
- name: cnr-data
  vld-id-ref: cnr-data
  floating-ip-required: true
- name: cnr-control
  vld-id-ref: s2
- name: cnr-public
  floating-ip-required: true
  vld-id-ref: cnr-public
- name: cinar-public
  floating-ip-required: true
  vld-id-ref: cinar-public
```

```
ip-profiles:
- name: s1
  description: s1 network
  ip-profile-params:
    ip-version: ipv4
    subnet-address: 13.10.21.0/24
    dhcp-params:
      enabled: true
- name: s2
  description: s2 network
  ip-profile-params:
    ip-version: ipv4
    subnet-address: 14.10.21.0/24
    dhcp-params:
      enabled: true

vld:
- id: s2
  name: s2
  short-name: s2
  type: ELAN
  # mgmt-network: 'true'
  ip-profile-ref: s2
  vim-network-name: cinar-mgmt
  vnfd-connection-point-ref:
  - member-vnf-index-ref: 1
    vnfd-id-ref: mslice1_nrf_vnfd
    vnfd-connection-point-ref: cnr-control
    ip-address: 14.10.21.45
  - member-vnf-index-ref: 2
    vnfd-id-ref: mslice1_pcf_vnfd
    vnfd-connection-point-ref: cnr-control
    ip-address: 14.10.21.48
  - member-vnf-index-ref: 3
    vnfd-id-ref: mslice1_smf_vnfd
    vnfd-connection-point-ref: cnr-control
    ip-address: 14.10.21.49
```


ÇINAR 5GC VNFD Design



Open Source MANO

MAIN NAVIGATION

- Home
- PROJECT
- Overview
- Packages
 - NS Packages
 - VNF Packages
- NetSlice Templates
- Instances
 - NS Instances
 - VNF Instances
 - PDU Instances
 - NetSlice Instances
- SDN Controllers
- VIM Accounts
- ADMIN
- Users
- Projects

VNF Packages

Show 10 entries

Short Name	Identifier	Type	Description
slice_amf_vnfd	226c12bf-eb53-40b4-b324-a6df8f44aa9c	vnfd	AMF with 3 interfaces (public, control, management)
slice_ausf_vnfd	75a5bfde-45b2-4bbd-88a4-a956be5e3ea7	vnfd	AUSF with 3 interfaces (public, control, management)
slice_nrf_vnfd	f177f485-e8a0-4632-9fab-35b4e5c7f259	vnfd	NRF with 3 interfaces (public, control, management)
slice_pcf_vnfd	5aa97a06-a121-43af-b914-d1aefdd51040	vnfd	PCF with 3 interfaces (public, control, management)
slice_smf_vnfd	73490edc-b8e4-4a65-8628-f0ab26400be3	vnfd	SMF with 3 interfaces (public, control, management)
slice_udm_vnfd	b7ecd307-2b95-4f70-b9b0-e3f462da0f3	vnfd	UDM with 3 interfaces (public, control, management)
slice_udr_vnfd	a991695e-acc2-42dd-a285-d9b7a26b1daf	vnfd	UDR with 3 interfaces (public, control, management)
slice_upf_vnfd	01cd2dc9-7044-4b1c-ae7a-58530fc4be17	vnfd	UPF with 3 interfaces (public, control, management)

```

connection-point:
- name: cnr-control
  type: VPORT
- name: cinar-mgmt
  type: VPORT
- name: cinar-public
  type: VPORT
    
```

```

vnf-configuration:
  metrics:
    - name: users
    - name: load
    - name: load_pct
    - name: cpu
    - name: memory
  initial-config-primitive:
    - seq: '1'
      name: config
      parameter:
        - name: ssh-hostname
          value: <rw_mgmt_ip>
        - name: ssh-username
          value: cnrusr
        - name: ssh-password
          value:
    - seq: '2'
      name: installdeb
    - seq: '3'
      name: cnrconfig
    
```

```

juju:
  charm: udrconf1
    
```

```

monitoring-param:
- id: "ubuntu_users"
  name: "ubuntu_users"
  aggregation-type: AVERAGE
  vnf-metric:
    vnf-metric-name-ref: "users"
- id: "ubuntu_load"
  name: "ubuntu_load"
  aggregation-type: AVERAGE
  vnf-metric:
    vnf-metric-name-ref: "load"
- id: "ubuntu_load_pct"
  name: "ubuntu_load_pct"
  aggregation-type: AVERAGE
  vnf-metric:
    vnf-metric-name-ref: "load_pct"
- id: "cpu_usage"
  name: "cpu_usage"
  aggregation-type: COUNT
  vnf-metric:
    vnf-metric-name-ref: "cpu"
- id: "memory_usage"
  name: "memory_usage"
  aggregation-type: COUNT
  vnf-metric:
    vnf-metric-name-ref: "memory"
    
```

ETSI Plugtest Experience



Collaborations

MANO

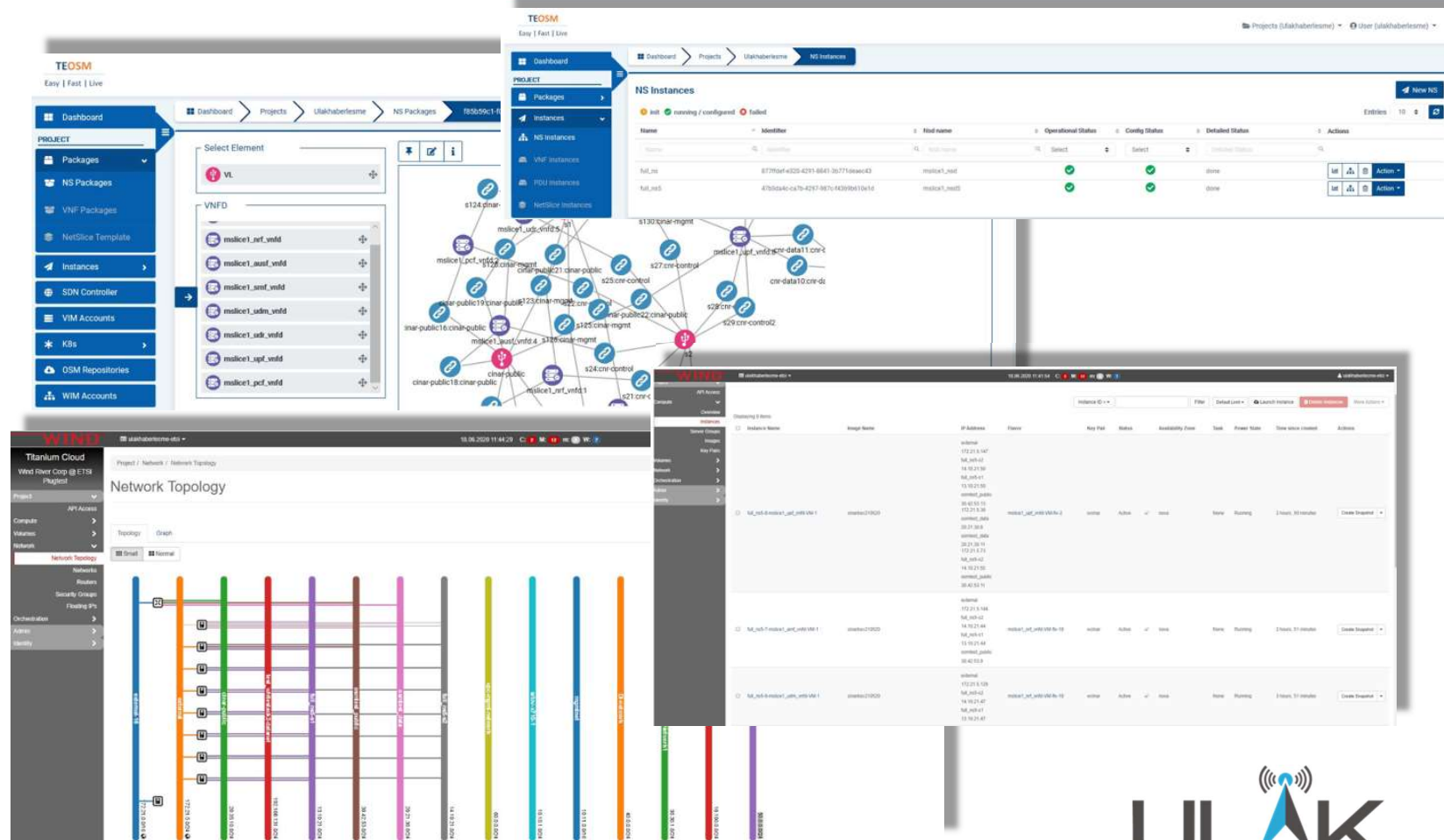
Whitestack-WhiteNFV
TATA Elxsi- TEOSM

NFVI-VIM

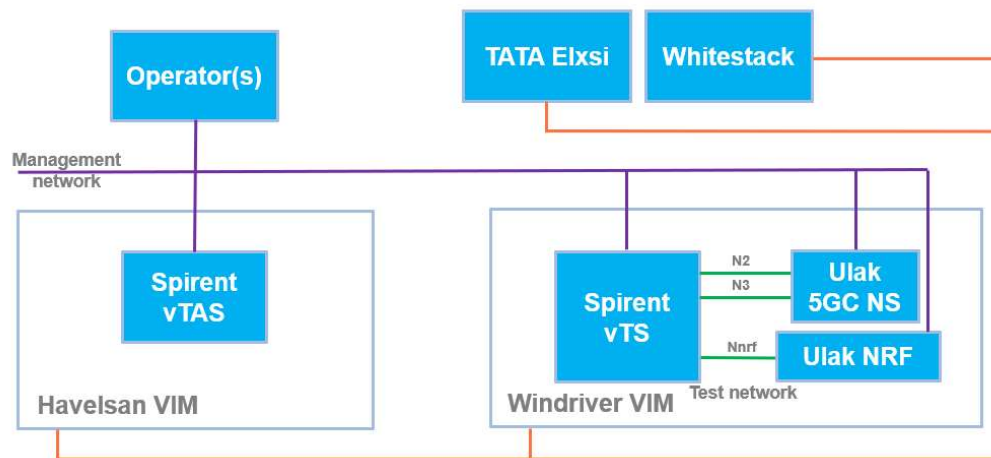
Windriver
Whitestack-Whitecloud
Havelsan-Telco Cloud

VNF Vendors

Ulak
Spirent



ETSI Plugtest Experience



5G Use Case Demo: OSM Orchestrated 5G-related NS NRF testing scenario:

Ulak's NRF functionality has been verified by the Landslide capabilities of AMF emulation for NF registration and SMF discovery to ULAK's NRF

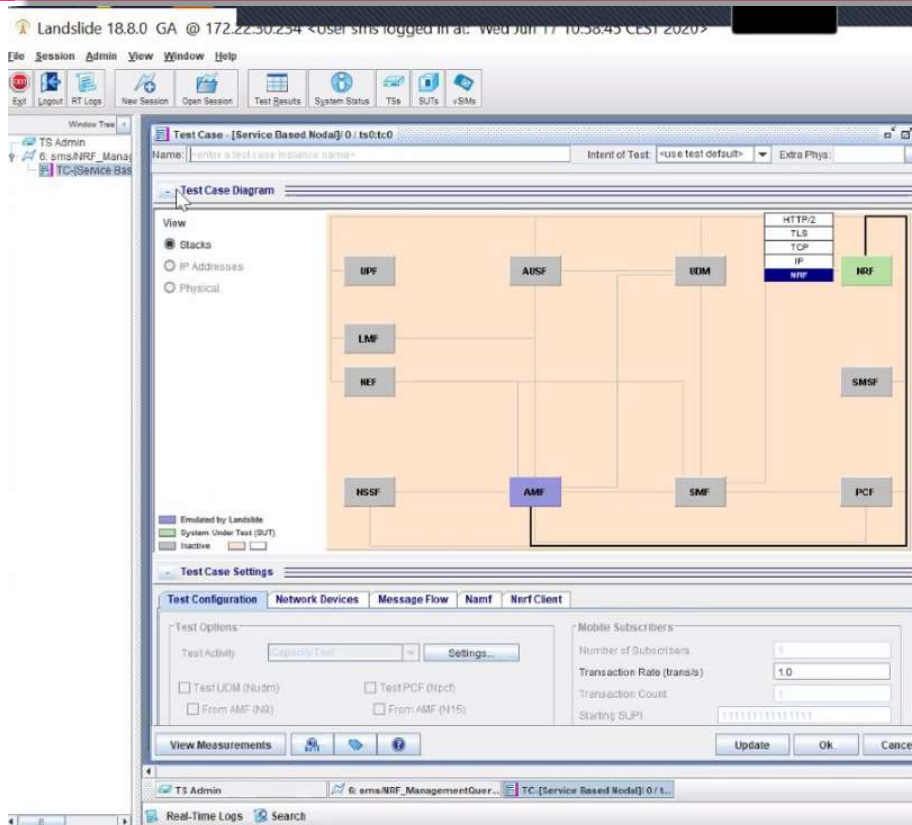
5GC E2E testing scenario:

Registration and PDU session establishment to ULAK's 5GC has been successfully completed by Landslide's 5G RAN emulation capability for Ulak VNF's

Demo Setup

- Ulak NS includes 9 VNFs
- Spirent Landslide is composed of 2 NS's: vTAS and vTS
- ULAK's NRF and Landslide vTS were deployed on Windriver via Whitestack
- ULAK's 5GC was deployed on Windriver VIM via Tata Elxsi
- Landslide vTAS NS was deployed on Havelsan VIM

ETSI Plugtest Experience



sp Vision Powered by

Test Session: 6: smsNRF_ManagementQuery

Overall State: 2_DELAY

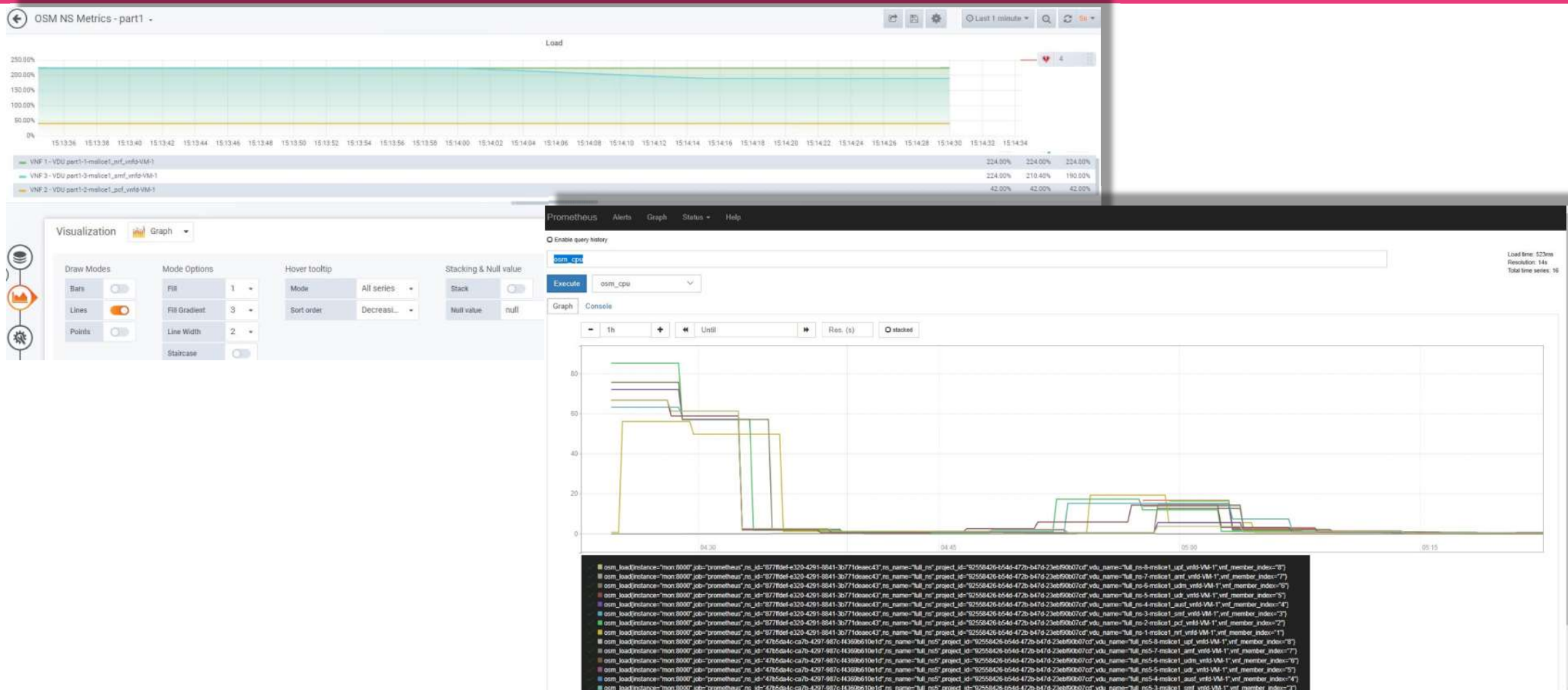
Session Builder Automation Control Pass/Fail Port Capture Reports Favorites Info Logs

Options: Filters Enable Filtering Interval: 12 of 12 Go to Interval Go

Report View: Summary

Measurement	AMF SBI Node	AMF Server TCP	AMF Server HTTP2	Nrf Client TCP	Nrf Client HTTP2	Nrf Consumer	Current
Classed Time	2 Min(s) 15 Sec(s)	2 Min(s) 30 Sec(s)	2 Min(s) 45 Sec(s)	3 Min(s) 0 Sec(s)	3 Min(s) 0 Sec(s)	3 Min(s) 4 Sec(s)	
Actual Time	06/17 12:00:46	06/17 12:01:01	06/17 12:01:16	06/17 12:01:31	06/17 12:01:35	06/17 12:01:35	
NF Complete Update Requests	2	2	2	2	2	2	
NF Complete Update Successes	2	2	2	2	2	2	
NF Heartbeat Requests	4	4	5	5	5	5	
NF Heartbeat Successes	4	4	5	5	5	5	
NF List Retrieval Requests	2	2	3	3	3	3	
NF List Retrieval Successes	2	2	3	3	3	3	

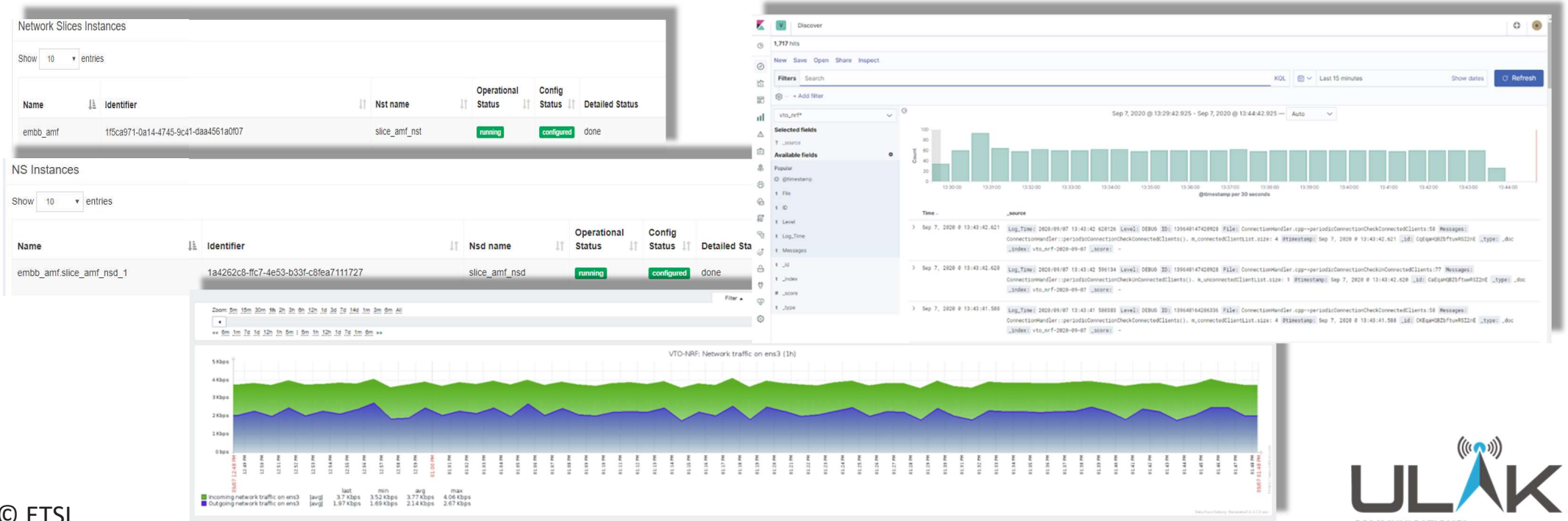
ETSI Plugtest Experience



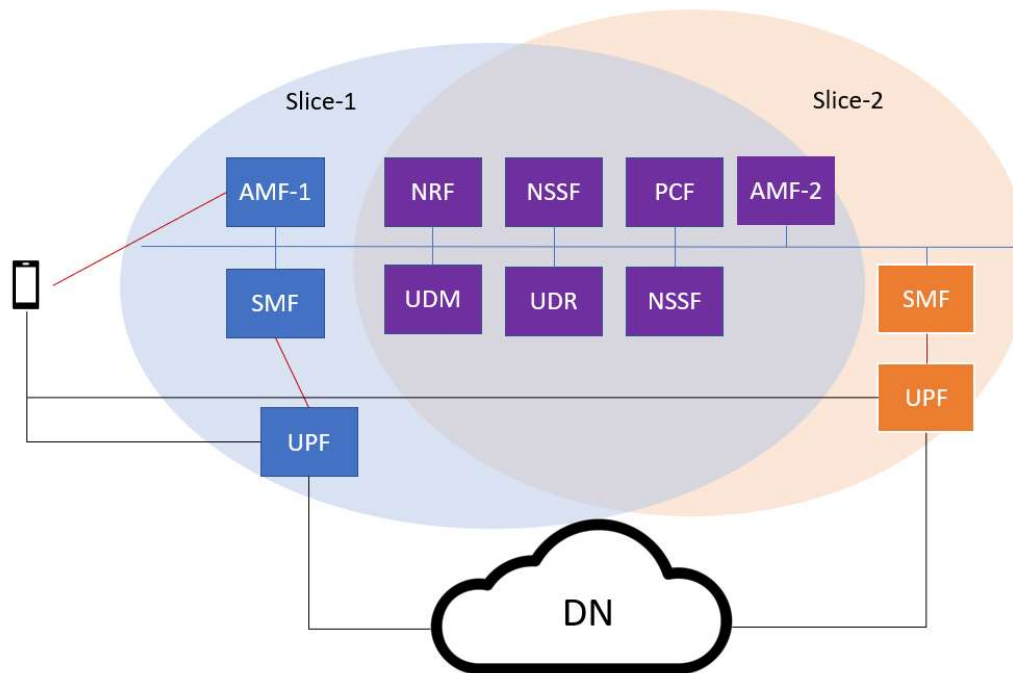
OSM Usage



- VNF configuration and recovery with day-2 actions
- 5G Core network orchestration
- Manuel scaling
- Performance monitoring and alarm creating by Zabbix integration
- Log collection and error reporting by integration with Elasticsearch - Fluentd - Kibana (EFK)



Plans and Comments

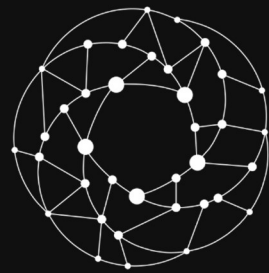


Plan

- Use OSM to create network slices within separate and shared NS's
- Autoscale the VNF's

Difficulties Experienced

- Configuration of scaled VNFs for a dynamic scenario
- Getting notifications from OSM in status changes



Open Source
MANO

THANKS

