OSM White Paper

OSM RELEASE THIRTEEN RELEASE NOTES

December 2022

Open Source MANO

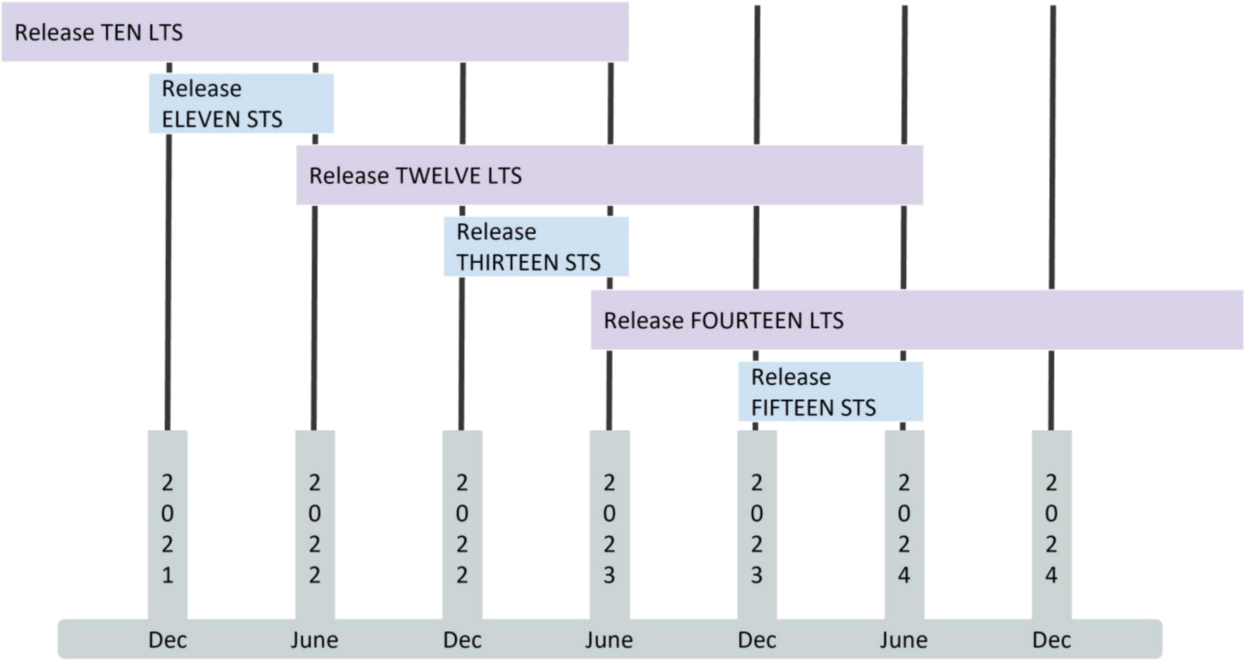
Technical Steering Committee

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

Contents

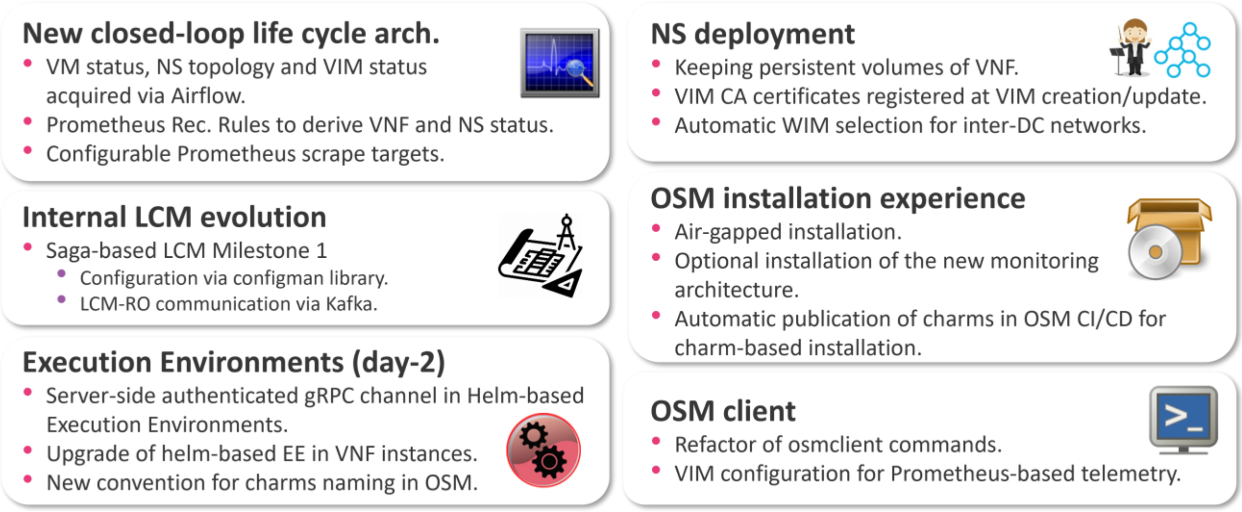
Introduction

The ETSI Open Source MANO community is proud to announce OSM Release THIRTEEN, meeting our already established cadence of two releases per year, alternating between LTS releases such as Release TWELVE (2 years support) and Standard releases (6 months support).



OSM release cadence

Release THIRTEEN includes significant improvements in many key areas.



New closed-loop life cycle arch.

- VM status, NS topology and VIM status acquired via Airflow.
- Prometheus Rec. Rules to derive VNF and NS status.
- Configurable Prometheus scrape targets.

Internal LCM evolution

- Saga-based LCM Milestone 1
  - Configuration via configman library.
  - LCM-RO communication via Kafka.

Execution Environments (day-2)

- Server-side authenticated gRPC channel in Helm-based Execution Environments.
- Upgrade of helm-based EE in VNF instances.
- New convention for charms naming in OSM.

NS deployment

- Keeping persistent volumes of VNF.
- VIM CA certificates registered at VIM creation/update.
- Automatic WIM selection for inter-DC networks.

OSM installation experience

- Air-gapped installation.
- Optional installation of the new monitoring architecture.
- Automatic publication of charms in OSM CI/CD for charm-based installation.

OSM client

- Refactor of osmclient commands.
- VIM configuration for Prometheus-based telemetry.

Release THIRTEEN - Feature summary

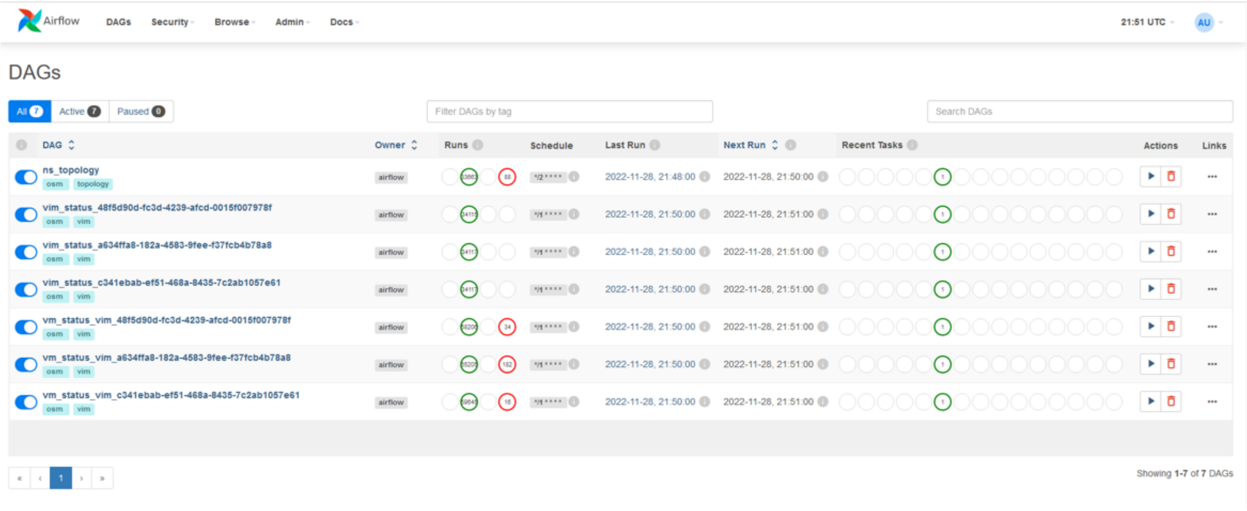
New architecture for closed-loop operations

This release introduces a new scalable architecture for service assurance and closed-loop operations leveraging on cloud-native version of Apache Airflow and Prometheus. This architecture is prepared to cover the most demanding service assurance scenarios such as auto-healing and auto-scaling in large clouds and multiple edge sites. Release THIRTEEN incorporates new workflows for getting the state of Network Functions (NF), Network Services (NS) and VIM, and will gradually incorporate new capabilities in the next releases.

Airflow monitoring pipelines for VM status, NS topology and VIM status

New monitoring workflows have been implemented to retrieve VM status, NS topology and VIM status by means of Airflow Directed Acyclic Graphs (DAGs):

- DAGs for VIM status and VM status are dynamically created depending on the number of VIM accounts. They have been designed to perform a single query to each VIM, thus reducing the interaction and reducing the resource consumption in comparison to the former MON architecture.
- DAG for NS topology takes the information from MongoDB and populates the topology in a convenient format into Prometheus TSDB, adding an entry for each VM with labels for the corresponding VNF and NS, which will allow later correlation of this metric with other metrics to derive advanced metrics.



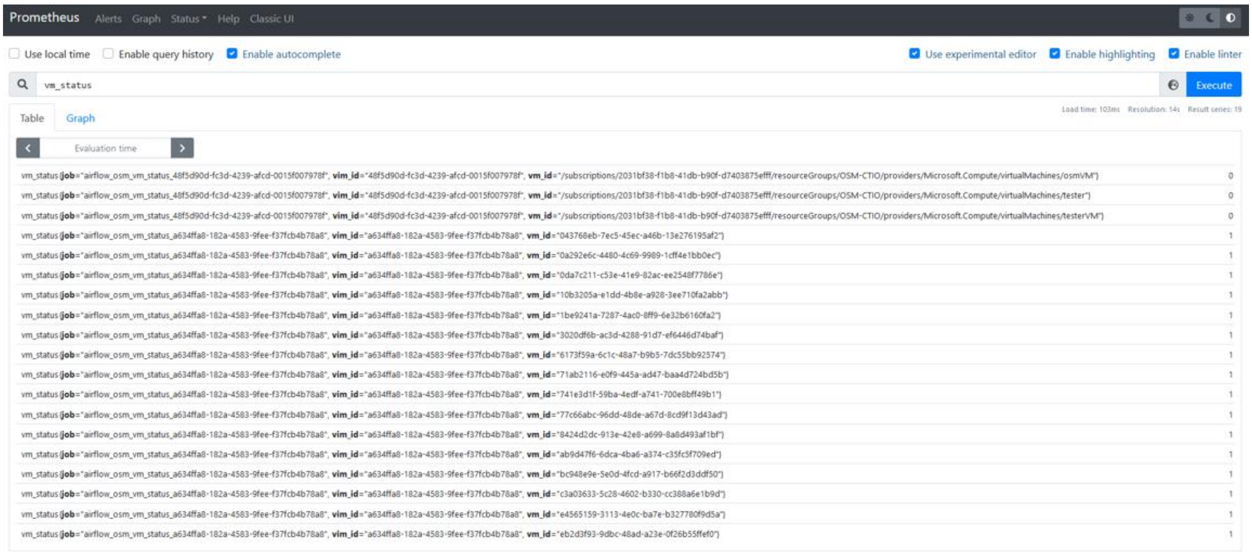
DAGs

DAG	Owner	Runs	Schedule	Last Run	Next Run	Recent Tasks	Actions	Links
ns_topology osm topology	airflow	33663 88	*/2 * * * *	2022-11-28, 21:48:00	2022-11-28, 21:50:00	1	▶ 🗑	···
vim_status_48f5d90d-fc3d-4239-afcd-0015f007978f osm vim	airflow	34115	*/1 * * * *	2022-11-28, 21:50:00	2022-11-28, 21:51:00	1	▶ 🗑	···
vim_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8 osm vim	airflow	34117	*/1 * * * *	2022-11-28, 21:50:00	2022-11-28, 21:51:00	1	▶ 🗑	···
vim_status_c341ebab-ef51-468a-8435-7c2ab1057e61 osm vim	airflow	34117	*/1 * * * *	2022-11-28, 21:50:00	2022-11-28, 21:51:00	1	▶ 🗑	···
vm_status_vim_48f5d90d-fc3d-4239-afcd-0015f007978f osm vim	airflow	58206 34	*/1 * * * *	2022-11-28, 21:50:00	2022-11-28, 21:51:00	1	▶ 🗑	···
vm_status_vim_a634ffa8-182a-4583-9fee-f37fcb4b78a8 osm vim	airflow	55205 182	*/1 * * * *	2022-11-28, 21:50:00	2022-11-28, 21:51:00	1	▶ 🗑	···
vm_status_vim_c341ebab-ef51-468a-8435-7c2ab1057e61 osm vim	airflow	59645 16	*/1 * * * *	2022-11-28, 21:50:00	2022-11-28, 21:51:00	1	▶ 🗑	···

Showing 1-7 of 7 DAGs

Airflow DAGs for VIM status, VM status and NS topology

In all the cases, the resulting metrics are pushed from the Airflow DAGs to Prometheus.



□ Use local time □ Enable query history ☑ Enable autocomplete ☑ Use experimental editor ☑ Enable highlighting ☑ Enable linter

vm_status

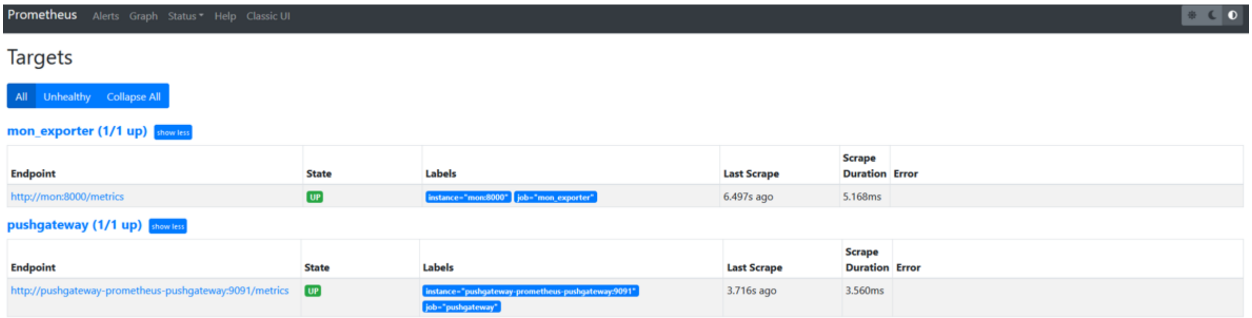
Load time: 103ms Resolution: 14s Result series: 19

```
vm_status{job="airflow_osm_vm_status_48f5d90d-fc3d-4239-afcd-0015f007978f", vim_id="48f5d90d-fc3d-4239-afcd-0015f007978f", vm_id="/subscriptions/2031bf38-f1b8-41db-b90f-d7403875efff/resourceGroups/OSM-CTIO/providers/Microsoft.Compute/virtualMachines/osmVM"}   0
vm_status{job="airflow_osm_vm_status_48f5d90d-fc3d-4239-afcd-0015f007978f", vim_id="48f5d90d-fc3d-4239-afcd-0015f007978f", vm_id="/subscriptions/2031bf38-f1b8-41db-b90f-d7403875efff/resourceGroups/OSM-CTIO/providers/Microsoft.Compute/virtualMachines/tester"}   0
vm_status{job="airflow_osm_vm_status_48f5d90d-fc3d-4239-afcd-0015f007978f", vim_id="48f5d90d-fc3d-4239-afcd-0015f007978f", vm_id="/subscriptions/2031bf38-f1b8-41db-b90f-d7403875efff/resourceGroups/OSM-CTIO/providers/Microsoft.Compute/virtualMachines/testerVM"}   0
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="043768eb-7ec5-45ec-a46b-13e276195af2"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="0a292e6c-4480-4c69-9989-1cff4e1bb0ec"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="0da7c211-c53e-41e9-82ac-ee2548f7786e"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="10b3205a-e1dd-4b8e-a928-3ee710fa2abb"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="1be9241a-7287-4ac0-8ff9-6e32b6160fa2"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="3020df6b-ac3d-4288-91d7-ef6446d74baf"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="6173f59a-6c1c-48a7-b9b5-7dc55bb92574"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="71ab2116-e0f9-445a-ad47-baa4d724bd5b"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="741e3d1f-59ba-4edf-a741-700e8bff49b1"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="77c66abc-96dd-48de-a67d-8cd9f13d43ad"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="8424d2dc-913e-42e8-a699-8a8d493af1bf"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="ab9d47f6-6dca-4ba6-a374-c35fc5f709ed"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="bc948e9e-5e0d-4fcd-a917-b66f2d3ddf50"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="c3a03633-5c28-4602-b330-cc388a6e1b9d"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="e4565159-3113-4e0c-ba7e-b327780f9d5a"}   1
vm_status{job="airflow_osm_vm_status_a634ffa8-182a-4583-9fee-f37fcb4b78a8", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="eb2d3f93-9dbc-48ad-a23e-0f26b55ffef0"}   1
```

VM status

Enabling addition of external targets to OSM Prometheus

Metrics generated from the previous DAGs are pushed to Prometheus by means of a Prometheus Pushgateway, which integrates seamlessly with the existing OSM Prometheus thanks to this new feature that allows to configure additional Prometheus scrape targets at will.



Targets

mon_exporter (1/1 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://mon:8000/metrics	UP	instance="mon:8000" job="mon_exporter"	6.497s ago	5.168ms	

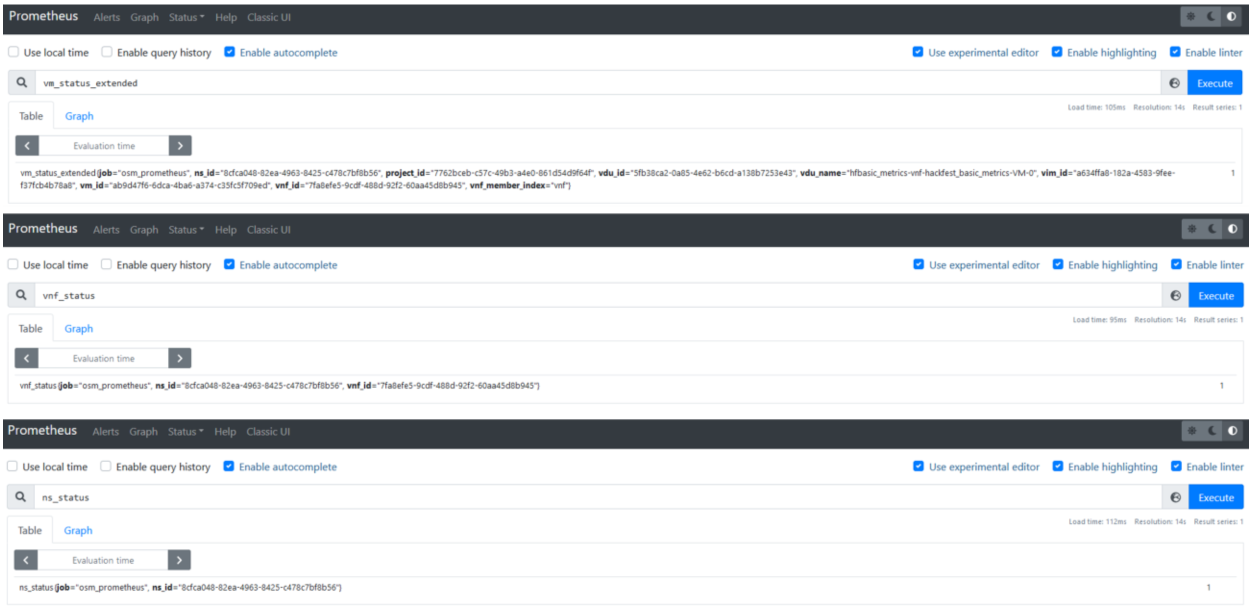
pushgateway (1/1 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://pushgateway-prometheus-pushgateway:9091/metrics	UP	instance="pushgateway-prometheus-pushgateway:9091" job="pushgateway"	3.716s ago	3.560ms	

OSM Prometheus scrape targets

Prometheus recording rules for VNF and NS status

From the previously acquired VM status and NS topology, Prometheus Recording Rules are used to derive new metrics such as the VNF and the NS status, which are conceptually equivalent to the existing metrics for VNF status and NS obtained with the former MON architecture, but without the performance penalties and the scalability issues present in the former MON architecture.



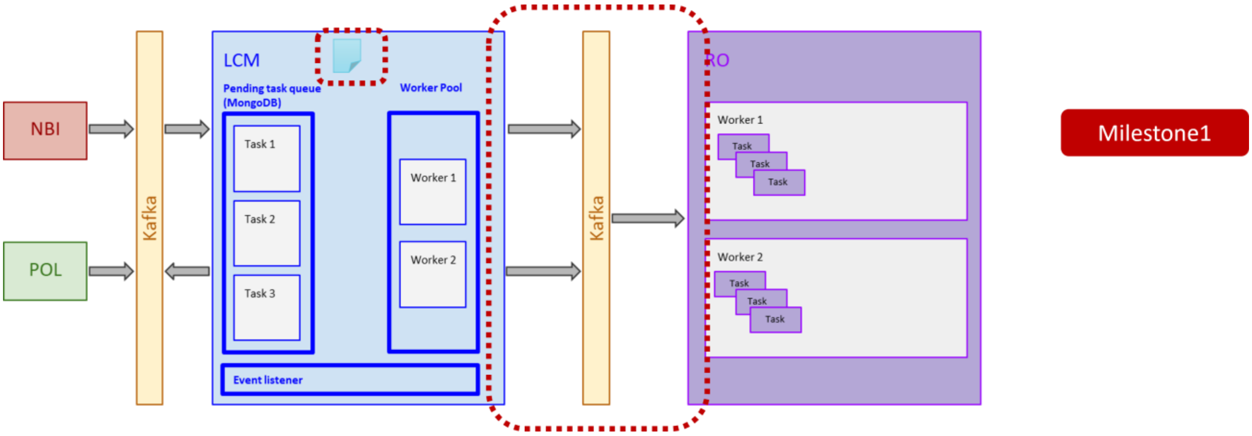
```
vm_status_extended
Load time: 105ms   Resolution: 14s   Result series: 1
vm_status_extended{job="osm_prometheus", ns_id="8cfca048-82ea-4963-8425-c478c7bf8b56", project_id="7762bceb-c57c-49b3-a4e0-861d54d9f64f", vdu_id="5fb38ca2-0a85-4e62-b6cd-a138b7253e43", vdu_name="hfbasic_metrics-vnf-hackfest_basic_metrics-VM-0", vim_id="a634ffa8-182a-4583-9fee-f37fcb4b78a8", vm_id="ab9d47f6-6dca-4ba6-a374-c35fc5f709ed", vnf_id="7fa8efe5-9cdf-488d-92f2-60aa45d8b945", vnf_member_index="vnf"}   1

vnf_status
Load time: 95ms   Resolution: 14s   Result series: 1
vnf_status{job="osm_prometheus", ns_id="8cfca048-82ea-4963-8425-c478c7bf8b56", vnf_id="7fa8efe5-9cdf-488d-92f2-60aa45d8b945"}   1

ns_status
Load time: 112ms   Resolution: 14s   Result series: 1
ns_status{job="osm_prometheus", ns_id="8cfca048-82ea-4963-8425-c478c7bf8b56"}   1
```

Derived metrics for VNF and NS status

Those basic indicators for VNF status and NS status will be the basis for building scalable closed loops for self-healing.

Internal LCM evolution

LCM module in OSM is responsible of managing the workflows associated to life cycle events of VNF and NS such as instantiation, termination, scaling, healing, and upgrading. In this Release, we have initiated the adoption of the Saga-based pattern for workflow management in selected operations. In addition, this release incorporates the communication channel between LCM and RO via Kafka.



LCM evolution - First milestone

New capabilities for NS deployment

OSM has proved to be successful in production environments in the deployment of NS. This release incorporates new capabilities for NS deployment such as the capability to make persistent volumes of NF permanent in the VIM, the ability to store CA certificates as part of VIM registration and update, or the automatic WIM selection for inter-DC networks.

NF volume persistence

With this new feature, it is possible now to make NF volumes persistent in the VIM after removing the NF. The intent to persist the volume can be specified in the descriptor or at instantiation time via the instantiation parameters. Below an example of the relevant excerpt of the VNF descriptor where the volume persistence is specified:

```
virtual-storage-desc:
- id: root-volume
  type-of-storage: persistent-storage
  size-of-storage: 10
  vdu-storage-requirements:
  - key: keep-volume
    value: 'true'
- id: persistent-volume
  type-of-storage: persistent-storage
  size-of-storage: 1
  vdu-storage-requirements:
  - key: keep-volume
    value: 'true'
```

VIM CA certificates

So far, the CA certificates required to interact with a VIM had to be added manually to RO containers. With this feature, the CA certificates are specified at VIM registration/update time, as follows:

```
osm vim-create --name openstack-site \
               --user admin \
               --password passwd \
               --auth_url https://10.10.10.12:5000/v3 \
               --tenant admin \
               --account_type openstack \
               --ca_cert /home/ubuntu/root_ca.pem
```

Automatic WIM selection for inter-datacenter networks

With this feature, the instantiation workflow has been updated to take into consideration the different scenarios of interconnection between sites through one or several WIM (WAN Infrastructure Manager).

Basically, a user can specify that VNF1 lands in VIM1 and VNF2 lands in VIM2. Depending on the existence of a WIM and the instantiation parameters, the end user can control the WIM that will be used:

1. If there is no WIM interconnecting VIM1 and VIM2, the NF will be deployed and connected only to the VIM-local virtual networks
2. If there is a registered WIM in OSM interconnecting VIM1 and VIM2, that WIM will be used to interconnect the VIM virtual networks.
3. If more than one WIM is registered in OSM and is interconnecting VIM1 and VIM2, the end user can indicate the WIM to be used.
4. Finally, the end user can disable the use of a WIM if desired by setting the instantiation parameter *wim_account* to False.

Other features

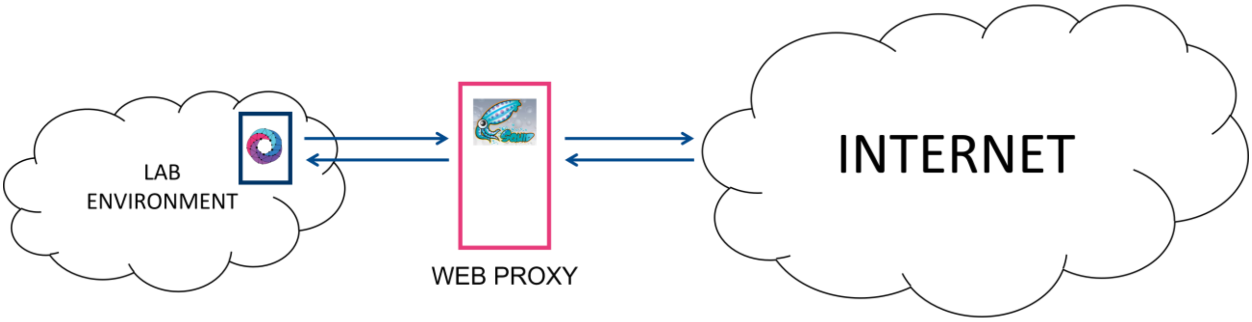
Improvements in Execution Environments

Day-2 operations are a key capability in OSM, which provides added value to the mere deployment of NS and NF through the use of Execution Environments (EE). The Execution Environments are isolated environments that allow the execution of code to perform operations on the NF and NS.

Release THIRTEEN provides an improved secured communication channel between helm-based EE and NF, the capability to upgrade helm-based EE and a new naming convention for Juju application in Juju-based EE.

Better OSM installation experience

The community installer will now be able to auto-detect installations of OSM behind a web proxy and perform the appropriate configurations, easing the global installation experience.



OSM Air-gapped installation behind a proxy

In addition, the internal process in OSM to generate the installation SW has incorporated the automatic publication of charms for charm-based installation.

Finally, the new scalable architecture for service assurance can be installed as an experimental option --*ng-sa* in the community installer.

Enhancements in the OSM client

Finally, this release includes some improvements in the OSM client, such as:

- a more convenient registration of Prometheus-based telemetry systems as part of VIM registration and update,
- the refactoring of OSM client commands,
- and an installation procedure for OSM client in Windows and Red Hat Linux distros in addition to Ubuntu Linux distros.

ETSI
06921 Sophia Antipolis CEDEX, France
Tel +33 4 92 94 42 00
info@etsi.org
www.etsi.org

This White Paper is issued for information only. It does not constitute an official or agreed position of ETSI, nor of its Members. The views expressed are entirely those of the author(s).

ETSI declines all responsibility for any errors and any loss or damage resulting from use of the contents of this White Paper.

ETSI also declines responsibility for any infringement of any third party's Intellectual Property Rights (IPR), but will be pleased to acknowledge any IPR and correct any infringement of which it is advised.