

Open Source
MANO

by ETSI

OSM RELEASE TWELVE

RELEASE NOTES

OPEN SOURCE MANO

TECHNICAL STEERING COMMITTEE

JUNE 2022

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




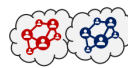




Contents

Contents	2
Introduction	3
New features in Release TWELVE	5
Network Function Healing	5
ETSI NFV-SOL003 support	6
Improved NS and NF lifecycle management	6
Additional support for Anti-Affinity	9
Extensions for CNF/Kubernetes support	9
Other enhancements	10
Conclusions	10
About OSM	11

Introduction

The ETSI Open Source MANO community is proud to announce OSM Release TWELVE, the second LTS release of ETSI OSM, providing two years of continuous support with bug fixes and security patches.

This is one of the most prolific releases of ETSI OSM, including significant improvements in many key areas.

<p>NF Healing</p> <ul style="list-style-type: none"> • VDU Healing • Auto-healing 	<p>Additional support for Anti-Affinity</p> <ul style="list-style-type: none"> • Anti-affinity groups per VNF 
<p>SOL003 support</p> <ul style="list-style-type: none"> • SOL003 support as G-VNFM • SOL003 subscription to VNF-LCM interface 	<p>Extensions for CNF/K8s support</p> <ul style="list-style-type: none"> • Day-2 primitives for CNF with proxy charms. • K8s cluster registration options. 
<p>NS Lifecycle</p> <ul style="list-style-type: none"> • Upgrade of VNF instances (upgrade of charms and primitives). • Removal of VNF instance from running NS. 	<p>Security</p> <ul style="list-style-type: none"> • Enforce password change on first login • Expire password after preset number of days. 

One of the main novelties is the capability to heal VNFs. Operators can now decide whether to manually heal a VNF or to model the VNF with healing policies to let the VNF automatically self-heal when any of its components is in an error state. This healing process can be applied in all the clouds supported by OSM, in line with OSM's multi-cloud approach

OSM Release TWELVE also brings significant improvements in the day-2 operation of a Network Service. This release adds the capability to upgrade a running VNF instance with new revisions of a VNF package. This allows not only the upgrade of a VNF with new versions of the images, but also the upgrade of the execution environments and primitives used by a VNF, with no impact on the running instance. This greatly improves the capability to operate a VNF since now both entities (the VNF itself and the SW operating it from OSM) can be upgraded independently. Moreover, the composition of network services can also be modified at runtime.

To further increase interoperability with 3rd party solutions, ETSI OSM Release TWELVE supports additional ETSI NFV Specifications, such as ETSI [GS NFV-SOL003](#) for VNF Lifecycle Management. With the support of the VNF Lifecycle Management APIs, Release TWELVE offers both the API of an NFV Orchestrator and the API of a Generic VNF Manager, providing operators with additional options for the integration of OSM in their Telco Cloud Architecture. This addition continues the alignment journey initiated at OSM creation, in which ETSI [GS NFV-SOL005](#) was adopted as the foundation of OSM's northbound interface, OSM data model and descriptors were derived from ETSI [GS NFV-SOL006](#), and OSM NF and NS package formats were also based on ETSI [GS NFV-SOL004](#) and ETSI [GS NFV-SOL007](#).

Finally, this new release includes other interesting features for production, like the support of anti-affinity groups per VNF (which has also been backported to Release TEN), the addition of password

expiration policies, new options for Kubernetes cluster registration, and the experimental migration of Openstack-based VM instances.

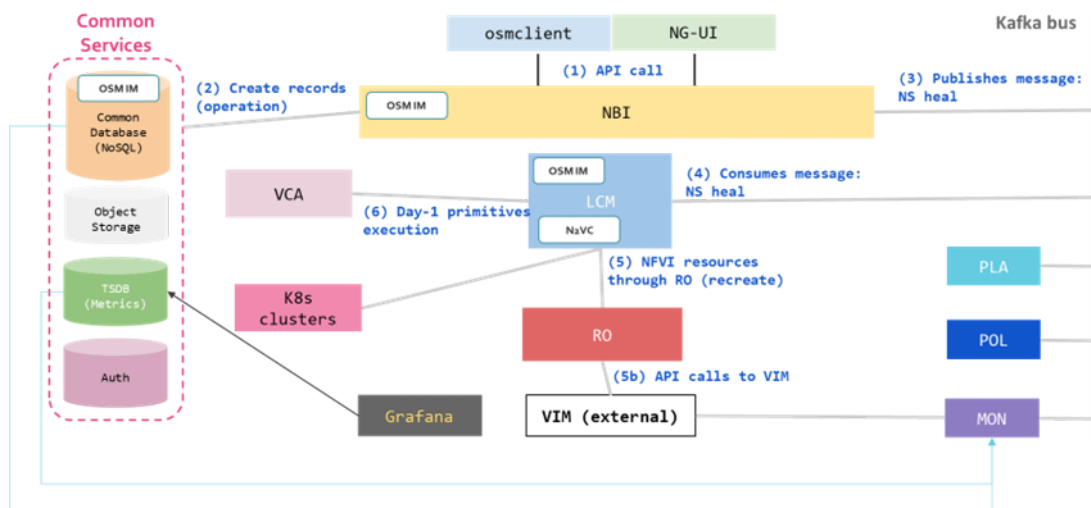
We were able to launch Release TWELVE with the effort and contribution of the organizations that integrate the OSM community. Many thanks to all these organizations, OSM developers, and users, participating directly or indirectly in making OSM awesome.

New features in Release TWELVE

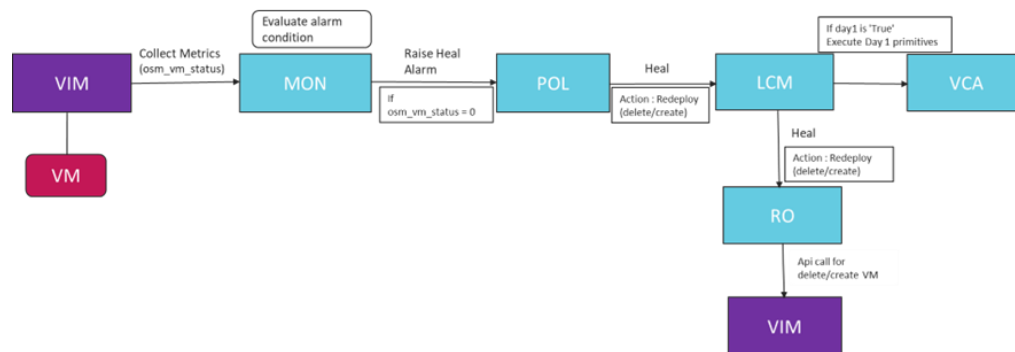
Network Function Healing

Release TWELVE brings the ability to heal network functions affected by any kind of infrastructure failure. This healing process, which may be triggered either manually or automatically depending on operator preference, can be applied in all the clouds supported by OSM, in line with OSM's multi-cloud approach.

Manual healing can be triggered through a new API call that has been added, following ETSI GS NFV-SOL005. This API call triggers the healing of one or several VNFs in a single call, allowing to indicate the specific VDUs to be healed, and optionally enabling the execution of day-1 primitives in the healed VDU and VNF. As a result, the impacted VM/VDU are recreated and properly initialized.



In addition, the capability to heal a VNF is complemented with a closed-loop mechanism, where the metrics coming from the VM can be used to trigger an automated healing. Thanks to this intent-driven mechanism in the information model, VNF designers can specify a new healing-aspect in the VNF descriptor, similar to the already existing scaling-aspect, indicating the list of VDUs to be re-deployed. When OSM detects that any of the VDUs is in error state, if the policy allows to raise an alarm, those VDUs are re-deployed.



ETSI NFV-SOL003 support

OSM has always allowed the lifecycle management of Network Functions and Network Services. However, up to now, the only Northbound Interface offered by OSM was the NFVO's one, based on [ETSI GS NFV-SOL005](#).

With this new feature, OSM incorporates a new set of API calls based on [GS NFV-SOL003](#). These API calls allow 3rd party solutions to manage the life cycle of Network Functions without the need of understanding the whole Network Service. Thanks to this feature, OSM can now play the role of an independent Generic-VNFM, of an NFV-Orchestrator, or both if desired.

Besides the addition of VNF lifecycle management API calls to instantiate, scale and terminate the NF, this feature is complemented with the capability to subscribe to those events and be notified, in the same way that it was already supported for NS lifecycle events. The whole list of new supported API calls can be found hereafter:

```
{apiRoot}/vnflcm/v1
  /vnf_instances
  /vnf_instances/{vnfInstanceId}
  /vnf_instances/{vnfInstanceId}/instantiate
  /vnf_instances/{vnfInstanceId}/scale
  /vnf_instances/{vnfInstanceId}/terminate
  /vnf_lcm_op_occs
  /vnf_lcm_op_occs/{vnfLcmOpOccId}
  /subscriptions/{subscriptionId}
```

Improved NS and NF lifecycle management

OSM Release TWELVE brings significant improvements in the day-2 operation of a Network Service. It adds the capability to upgrade a running NF instance to use a new version of the images, but also to upgrade the execution environments and primitives used by the NF with no impact on the running NF instance. This greatly improves the capability to operate a NF since now both entities (the NF itself and

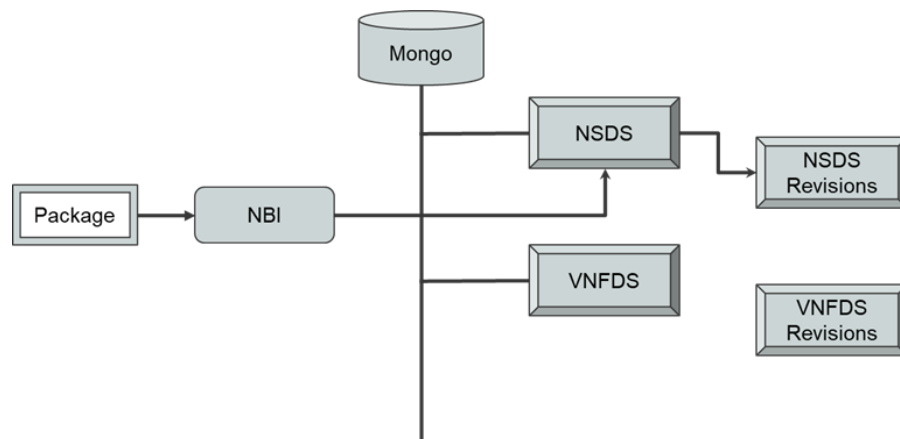
the SW operating it from OSM) can be upgraded independently. Moreover, VNF and its VDUs can be individually stopped, (re)started or rebuilt at will.

In addition, the composition of network services can also be modified at runtime to remove, for instance, a running NF instance from a NS.

All these capabilities are built on top of two technical enhancements:

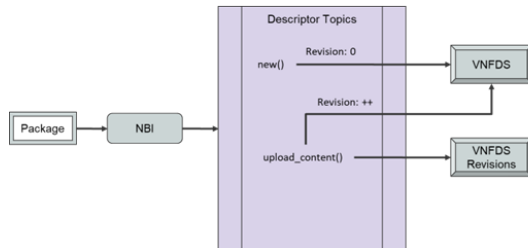
- The notion of revisions in NF and NS packages to model different versions of the same NF and NS resources
- A new umbrella API call to update a running NS instance following ETSI [GS NFV-SOL005](#), with the option to indicate the updateType, in order to execute the different life cycle operations (use of a new revision of a NF package, remove an existing NF instance from a running NS, start/stop/rebuild a VNF, etc.)

End users can now upload new versions of NF or NS packages, called revisions. Existing instances will continue to use the original version, but the new revisions will be used for the new instances.

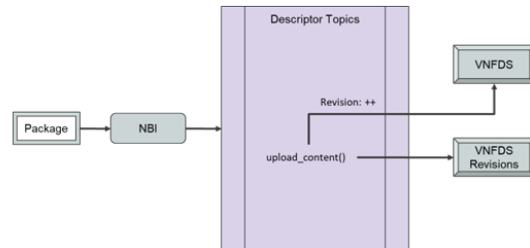


The different workflows to onboard packages, NS instantiation and NS termination were therefore adapted to be able to follow the notion of revision.

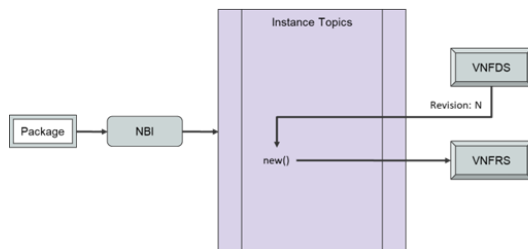
Flow for new package



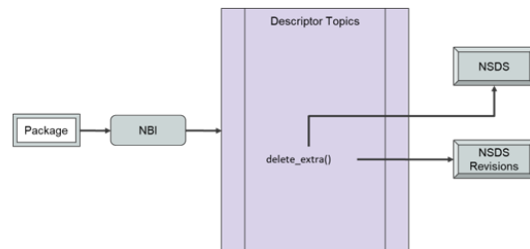
Flow for package update



Flow for NS instantiation



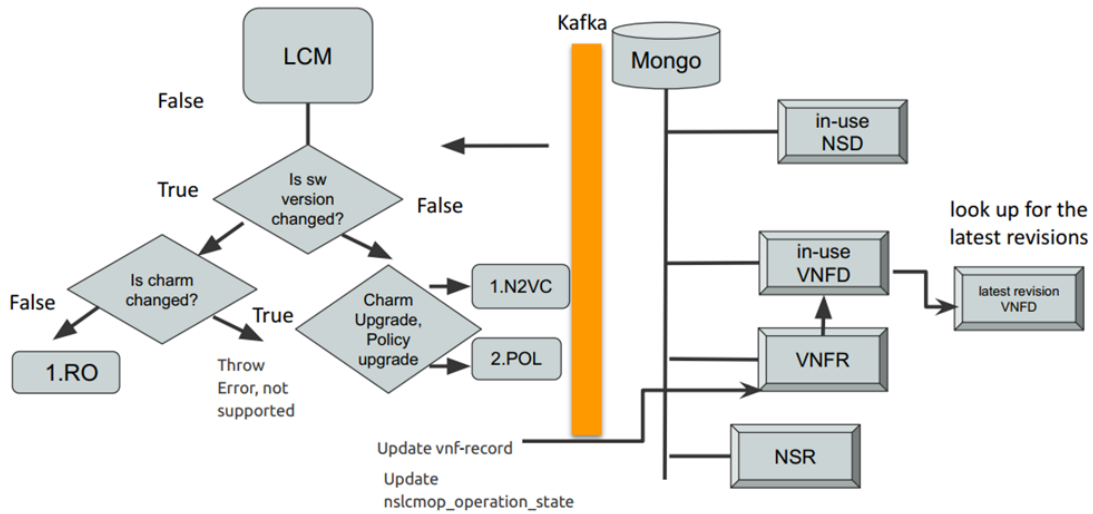
Flow for NS termination



In addition to the notion of revisions, a new API call was added to update a running NS instance following ETSI GS NFV-SOL005. The update API call is an umbrella API that allows different kinds of updates depending on the specified *updateType*, which allows the following use cases for the moment:

- Upgrade of a running NF instance to use a new revision of a NF package
- Upgrade of a running NF instance to use an updated descriptor including a new policy such as scaling or healing policies
- Remove an existing NF instance from a running NS
- Change the operational state of a NF or its VDU (start/stop/rebuild)

Depending on the update, different E2E workflows are triggered.



Additional support for Anti-Affinity

VNFs can now be deployed with High Availability, by declaring anti-affinity groups in the VNF descriptor to ensure that some VDUs are deployed on different hosts for redundancy purposes. The current implementation of this feature currently covers the management of those affinity and anti-affinity groups in Openstack-based environments (including VIO), and its scope might be extended to other clouds in the future.

This feature is descriptor-driven and follows the declarative modeling used in OSM: the definition of the affinity and anti-affinity groups, as well as the VDUs belonging to them, are included in the VNF descriptors.

Due to the importance of this feature for production environments, it has been backported to Release TEN.

Extensions for CNF/Kubernetes support

Release ELEVEN already brought the experimental support of Day-2 primitives for CNFs. Release TWELVE completes the remaining gaps, allowing now to fully exploit the configuration services offered by any of the Kubernetes Deployment Units (KDUs) of a CNF.

OSM can discover the Kubernetes services exposed by the different KDUs and deploy Execution Environments (for instance, proxy charms) that can execute day-2 primitives over those KDUs through the exposed services. In that way, CNFs offering a REST API or a Netconf API can be operated via an OSM Execution Environment. The Execution Environments and the KDUs managed by them are declared in the VNF descriptors, following the characteristic declarative modeling used in OSM.

In addition to this feature, Kubernetes cluster registration was also extended with additional options to restrict the initialization of the Kubernetes cluster to only Helm-based KDU or Juju-based KDU. In that way, the complexity of bootstrapping the Kubernetes cluster can be significantly reduced since only the methods specified by the operator will be applied.

Other enhancements

Release TWELVE comes also with other interesting features:

- Security enhancements to improve password management, like the enforcement of password changes on first login and a default policy for password expiration after a preset number of days
- The capability to monitor NFVI level metrics from Prometheus TSDB as Openstack telemetry system
- Vertical scaling of Openstack-based VM instances
- Migration of Openstack-based VM instances from OSM
- Improvements in the internal workflows for alarm notifications

Conclusions

Release TWELVE keeps our schedule of two releases per year since 2016. Moreover, this release renews our commitment for Long Term Support, becoming the second LTS release of ETSI OSM (after Release TEN), providing two years of continuous support with bug fixes and security patches.

This is one of the most prolific releases of ETSI OSM, including significant improvements in many areas: manual and automated NF healing, improved day-2 operation of a NS through upgrades of the NF and the SW operating it, support for anti-affinity groups, etc.

Committed since its foundation to the alignment with ETSI NFV standardization work, Release TWELVE also incorporates the adoption of ETSI GS NFV SOL003 specification for VNF LifeCycle Management, which allows OSM to offer both the API of an NFV Orchestrator and the API of a Generic VNF Manager, providing a wider range of options to operators for the integration of OSM in their Telco Cloud Architectures.

About OSM

ETSI OSM is a comprehensive open-source management and orchestration solution for physical, virtual and containerized Network Functions over private and public clouds, with three robust commercial distributions, and multiple commercial deployments by network operators.

The OSM community is still growing with ambitious plans. If you want to join us, please visit [How to get involved in OSM](#).

For more information, please visit the following links:

- [Install OSM Release TWELVE](#)
- [Open Source MANO User Documentation](#)
- [Open Source MANO VNF Onboarding Guide](#)
- [OSM Ecosystem](#)
- [OSM Members and Participants](#)