ETSI GR NFV-REL 011 V4.1.1 (2020-11)

Network Functions Virtualisation (NFV) Release 4; Management and Orchestration; Report on NFV-MANO software modification

Disclaimer

The present document has been produced and approved by the Network Functions Virtualisation (NFV) ETSI Industry Specification Group (ISG) and represents the views of those members who participated in this ISG.
It does not necessarily represent the views of the entire ETSI membership.

Reference

RGR/NFV-REL011ed411

Keywords

availability, MANO, NFV, service

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:
http://www.etsi.org/standards-search

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status. Information on the current status of this and other ETSI documents is available at
https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx

If you find errors in the present document, please send your comment to one of the following services:
https://portal.etsi.org/People/CommiteeSupportStaff.aspx

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

Foreword

This Group Report (GR) has been produced by ETSI Industry Specification Group (ISG) Network Functions Virtualisation (NFV).

Modal verbs terminology

In the present document "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document collects and describes use cases for modifying NFV-MANO software while maintaining service availability and continuity, irrespective of the technologies being used to deploy this software. As a result, detailed recommendations for the requirements of the NFV-MANO software modification process are derived.

2 References

2.1 Normative references

Normative references are not applicable in the present document.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the referenced document (including any amendments) applies.

NOTE: While any hyperlinks included in this clause were valid at the time of publication, ETSI cannot guarantee their long term validity.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] ETSI GS NFV-MAN 001 (V1.1.1): "Network Functions Virtualisation (NFV); Management and Orchestration".

[i.2] ETSI GS NFV-IFA 031 (V3.3.1): "Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Requirements and interfaces specification for management of NFV-MANO".

[i.3] ETSI GS NFV-IFA 009 (V1.1.1): "Network Functions Virtualisation (NFV); Management and Orchestration; Report on Architectural Options".

[i.4] ETSI GR NFV 003 (V1.5.1): "Network Functions Virtualisation (NFV); Terminology for main concepts in NFV".

[i.5] ETSI GR NFV-IFA 029 (V3.3.1): "Network Functions Virtualisation (NFV) Release 3; Architecture; Report on the Enhancements of the NFV architecture towards "Cloud-native" and "PaaS"".

[i.6] M. Toeroe and F. Tam (Eds.): "Service Availability: Principles and Practice", J. Wiley (2012).

[i.7] ETSI GS NFV-IFA 010 (V3.3.1): "Network Functions Virtualisation (NFV) Release 3; Management and Orchestration; Functional requirements specification".

[i.8] ETSI GR NFV-REL 007 (V1.1.2): "Network Functions Virtualisation (NFV); Reliability; Report on the resilience of NFV-MANO critical capabilities".

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI GR NFV 003 [i.4] and the following apply:

NOTE 1: A term defined in the present document takes precedence over the definition of the same term, if any, in ETSI GR NFV 003 [i.4].

NOTE 2: In the scope of the present document, for any entity which is part of the NFV architectural framework, the term "entity" is used.

fallback: recovery procedure of forcefully restoring an entity using a specific restoration point

NOTE 1: A successful fallback allows the entity to continue normal operation.

NOTE 2: A fallback may restore simultaneously more than one entity depending on the content of the restoration point.

NOTE 3: Depending on the content of the restoration point, the fallback operation does not ensure that the NFV-MANO functional entity continues to provide its services during the fallback procedure.

redundancy unit: part of an NFV-MANO functional entity which can have the option to be deployed redundantly to improve service availability

NOTE 1: When the redundancy unit(s) of an NFV-MANO functional entity is(are) deployed redundantly, they together with the rest of the NFV-MANO functional entity form a single instance of the NFV-MANO functional entity and, accordingly, act together to deliver better availability of the services provided by the NFV-MANO functional entity.

NOTE 2: The redundancy unit(s) of an NFV-MANO functional entity is(are) visible to the operator who takes the decision to deploy it(them) redundantly, or not, considering the supplier supported features.

restoration point: backup containing all data that is necessary to revert to an entity's software and state this entity had at the time the backup was captured

NOTE 1: A restoration point may contain data to restore more than one entity.

NOTE 2: A restoration point may be created by any techniques for capturing the software and the actual state of the entity. This can be a snapshot.

retry: graceful recovery procedure of re-entering the flow of the software modification process after the last successfully executed step prior to the error/failure occurrence

NOTE: A successful retry allows the continuation of the software modification process in progress as if no error/failure had occurred.

software modification: main part of the software modification process deploying the target software version

NOTE: Depending on the type of changes in the software deployed by a software modification, this part (or a subset) may be referred to as software upgrade or software update as defined in ETSI GR NFV 003 [i.4].

software modification process: process of replacing or modifying the software of one or more deployed functional entities

NOTE 1: The goal of the software modification process can be bug fixes or enhancements with or without adding, modifying or removing functionalities, interfaces or protocols.

NOTE 2: The software modification process may have two parts: software modification - (mandatory) main part deploying the intended target software version; and, if needed, software rollback - (conditional) recovery part restoring the software to the version deployed at the beginning of the software modification process if needed.

software rollback: graceful recovery procedure reversing a software modification

NOTE 1: A software rollback is undoing step by step the operations executed as part of the software modification. It can be triggered even after an operation resulted in an error.

NOTE 2: A software rollback restores the software to the version deployed at the beginning of the software modification, thereby ensuring that the involved NFV-MANO functional entities (i.e. targeted and impacted NFV-MANO functional entities) are in a consistent state (state synchronization might happen as part of this process).

NOTE 3: The intention is that during a software rollback, the NFV-MANO functional entity continues to provide its services.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

API	APplication Interface
CIS	Container Infrastructure Service
CISM	Container Infrastructure Service Management
EM	Element Manage
GS	Group Specification
LCM	Life Cycle Management
MCIO	Managed Container Infrastructure Object
NFVI	Network Functions Virtualisation Infrastructure
NFVO	Network Function Virtualisation Orchestrator
NS	Network Service
NSD	Network Service Descriptor
NSO	Network Service Orchestration
OS	Operating System
OSS	Operation Support System
RO	Resource Orchestration
SDN	Software Defined Networking
SMM	Software Modification Manager
VIM	Virtual Infrastructure Manager
VM	Virtual Machine
VNF	Virtual Network Function
VNFC	Virtual Network Function Component
VNFM	Virtual Network Function Manager
VRM	Virtual Resource Management
WAN	Wide Area Network
WIM	WAN Infrastructure Manager

4 Overview of NFV-MANO software modification

4.1 NFV-MANO software modification process

4.1.1 Introduction

As NFV-MANO is composed of different NFV-MANO functional entities, the software modification process consists of modifying the software of one or more of such functional entities. Two approaches for such modification are possible: online modification and offline modification. The online approach allows for in-service modification of the NFV-MANO functional entity(ies) software. As for the offline approach, it requires some downtime during the software modification. As service continuity is a prerequisite (see ETSI GR NFV-REL 007 [i.8]), the present document will only handle the online approach for NFV-MANO software modification process.

In addition to the online/offline distinction, the nature of the NFV-MANO functional entity(ies) has to be taken into consideration. Actually, the NFV-MANO functional entity(ies) can be deployed as VNF(s): in such case, its(their) software modification procedure is similar to the one considered for, e.g. network function VNFs. Another case is illustrated by NFV-MANO functional entity(ies) not deployed as VNF(s). Therefore, only the latter case will be discussed in the present document.

NOTE: Not being deployed as a VNF does not restrict that the NFV-MANO functional entity cannot be deployed as a virtualised entity; it only implies that it is not managed as a VNF.

Internal support of seamless software modification means that the creation of new NFV-MANO functional entity(ies) is not needed thanks to mechanisms embedded in the NFV-MANO functional entity(ies), e.g. redundant elements used for hosting the new software version. As such process is internal to the NFV-MANO functional entity(ies), the change of interface endpoints to the post-modification NFV-MANO functional entity(ies) is not necessary. Two approaches are thus possible: with or without internal support of seamless software modification. Both cases will also be handled in the present document.

Finally, it is noteworthy that two situations can occur: NFV-MANO functional entity running without redundancy, or cluster-like NFV-MANO functional entity based on an N+K architecture. In the latter case, the redundancy can be deployed locally in a single site or geographically across multiple sites.

4.1.2 Main concepts

As described in clause 4.1.1, the **software modification process** is the process of modifying or replacing the software of one or more functional entities deployed in the NFV-MANO, while the NFV-MANO continues to provide its services.

The goal of the software modification process is usually to introduce bug fixes or enhancements to the system or its subsystems with, or without, adding, modifying or removing functionalities, interfaces or protocols. To achieve these goals, the software modification process starts out with its main part - the **software modification**. During this software modification part, the software modification process takes actions progressively moving forward on the path towards deploying the intended or target software version. Depending on the type of changes in the software compared to the currently deployed software version, this part (or a subset) may be referred to as software upgrade or software update as defined in ETSI GR NFV 003 [i.4].

The software modification process itself is managed by an entity referred to as the Software Modification Manager (SMM). The SMM is responsible to coordinate the software modification process at the NFV-MANO system level by triggering the appropriate actions on potentially multiple different entities. These entities include the target NFV-MANO functional entities, but also other NFV-MANO functional entities and impacted entities, which might be managed by, or interacting with, the target NFV-MANO functional entities. Thus, the role of SMM can be fulfilled by an administrator or OSS, or it could be a new functionality offered by NFV-MANO. Considering this last option, it is important to point out the difference between the SMM and the internal support of an NFV-MANO functional entity for software modification.

The internal software modification support of an NFV-MANO functional entity is provided by its supplier and it manages the software modification of this NFV-MANO functional entity within its scope. The internal software modification support of an NFV-MANO functional entity is fully aware of the internal structure of this NFV-MANO functional entity. However, for the SMM, this internal software modification support is visible only as a single operation, which it can trigger on this NFV-MANO functional entity as part of the overall software modification process that can include other entities as well. When using the internal software modification, the SMM does not need to be aware of the internal structure of the NFV-MANO functional entity.

If the software modification of an NFV-MANO functional entity needs to be supported by the SMM, the SMM needs to be aware of parts of this NFV-MANO functional entity providing redundancy, that is, the **redundancy units** of the NFV-MANO functional entity, when the SMM needs to use this redundancy in the software modification process.

The assumption is that when redundancy units of an NFV-MANO functional entity are deployed redundantly, they act together as a single instance of the NFV-MANO functional entity to ensure better availability of the services provided by the NFV-MANO functional entity. The SMM can take advantage of such redundant deployment to improve service availability during the software modification process.

The actions the SMM typically orchestrates during the main software modification part are:

1) Preparation for the software modification, that is, performing different checks and operations to ensure as much as possible that the software modification will be successful, or if the software modification fails, then to ensure that a recovery is possible. The preparation includes checking and reserving resources, selecting or creating a **restoration point**, collecting information, starting special logs, etc.

2) One or more lifecycle management operations on the target NFV-MANO functional entities, which are necessary to deploy the target software. This can include triggering the internal software modification support of the target NFV-MANO functional entity, or the instantiation and termination of the target NFV-MANO functional entities or their redundancy units. Additional actions can be performed as necessary to manage the services provided by the target entities to their consumers.

3) Testing at different levels that the newly deployed software performs as expected and the software modification was successful. An NFV-MANO functional entity with internal support for software modification can perform testing within its scope, while the SMM is responsible to ensure that the software modification was successful at the system level.

4) Committing the changes introduced by the successful software modification.

It is envisioned that the details of the software modification process are provided to SMM by the operator as an execution plan, which is a workflow that the SMM can execute to achieve the intended software modification.

Unfortunately, there is no hundred percent guarantee that the software modification can be accomplished successfully, e.g. an action can fail persistently even after retries, or the target NFV-MANO functional entities with the newly deployed software may not pass all the required tests at the system level. This means that a software modification process also needs to include a conditional recovery part - the **software rollback**. A software rollback reverts the changes made by the software modification by undoing step by step the operations executed as part of the software modification. As a result, a software rollback restores the software to the source versions deployed at the beginning of the software modification, thereby ensuring that the involved entities (i.e. targeted and impacted entities) are in a consistent state (state synchronization process might happen as part of this process). The software rollback is a graceful recovery procedure as the goal is that the target NFV-MANO functional entities continue to provide their services during the procedure.

The software rollback part is also managed by the SMM using the execution plan in the same way as in the software modification part. It can also use information collected as part of the preparation to the software modification as well as during its execution, e.g. special logs supporting the software rollback.

The recovery part consists of actions similar to the software modification part:

1) Preparation for the software rollback, that is, performing different checks and collecting information to ensure that the software rollback is possible and will be successful.

2) One or more lifecycle management operations on target NFV-MANO functional entities necessary to revert them to their source software that was deployed at the beginning of the software modification. Additional actions can be performed as necessary to manage the services provided by the targeted entities to their consumers.

3) Testing that the re-deployed source software performs as before and the software rollback was successful.

4) Committing the changes made by the successful software rollback.

A software rollback can be triggered at any of the actions described in the software modification part which was performed, even after an erroneous operation, provided the system and specifically the target NFV-MANO functional entity are in a known state, i.e. a state from which graceful recovery is possible.

Depending on the target NFV-MANO functional entity and/or the action at which the problem occurred, it might be still possible to complete the software modification by re-trying the erroneous operation, i.e. re-entering the flow of the software modification process after the last successfully executed step prior to the error/failure occurrence. Moreover, **retry** can also be an option after an erroneous operation during a software rollback, which can avoid the need for a more drastic recovery procedure.

If neither retry nor rollback is possible, for example, because the system is in an unknown state, the last resort is a forceful recovery procedure, i.e. a **fallback**. The prerequisite of a fallback is that there is a restoration point available to which the target NFV-MANO functional entity and possibly other entities can be reverted so that normal operation can be continued.

The fallback procedure is not part of the software modification process, as failures or any other situation requiring such measures can occur outside of the scope of the software modifications process. However, the SMM can trigger a fallback to a given restoration point as a recovery procedure from a failed software modification process.

This **restoration point** is a backup containing all the data necessary to revert the software and the state of an entity (NFV-MANO functional entities and other entities) that this entity had at the time the backup was captured. The same restoration point can contain data to restore more than one entity, e.g. the target NFV-MANO functional entity, impacted entity, etc. Restoration points may be created by using different techniques for capturing the software and the actual state of entities such as a snapshot.

Figure 4.1.2-1 illustrates the relation between the different concepts related to the software modification process, while the next clause 4.1.3 elaborates further on the different recovery procedures introduced in this clause.

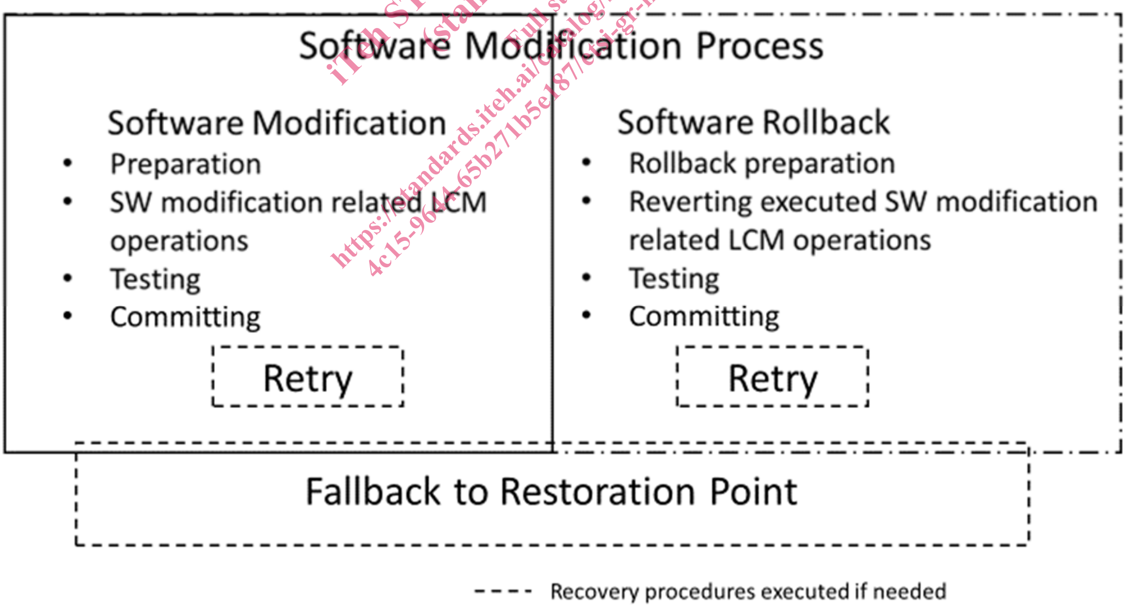

Figure 4.1.2-1: Interrelation of the software modification related concepts

4.1.3 Recovery procedures

From a resiliency perspective, the goal of the NFV-MANO software modification process is to modify the software of one or more NFV-MANO functional entities without impacting the services the NFV-MANO functional entities provide. Unfortunately, failures can occur at any time in the system including during, and as part of, the NFV-MANO software modification process. This clause introduces the different recovery procedures that can be applied during the software modification process.

M. Toeroe et al. [i.6] identify four artefacts determining the system state at any time. They apply to the NFV-MANO as follows:

1) The NFV-MANO configuration in terms of the NFV-MANO functional entities and their components.

2) The software executable associated with each of the NFV-MANO functional entity and their components.

3) The runtime status of each NFV-MANO functional entity and their components with respect to the services they provide.

4) The state of the service instances provided by the different NFV-MANO functional entities and their components.

At normal operation, when the NFV-MANO functional entities provide their service to the NFV system, typically only the last two artefacts (items 3 and 4) change, i.e. the runtime status of the NFV-MANO functional entities and the state of the service instances they provide. While at software modification the first two also change. Namely, the software of NFV-MANO functional entities is modified through manipulating the NFV-MANO configuration in such a way that it changes the software executable associated with the different NFV-MANO functional entities and their components, e.g. adding new NFV-MANO functional entities running the new software and removing old ones running the old software.

Accordingly, changes in items 1 and 2 together can be referred to as "software modification changes" and changes in items 3 and 4 referred to as "NFV-MANO service changes". The software modification process can also be defined as progressing with the software modification changes until they are successfully completed while not impacting the progress of the NFV-MANO service changes reflecting the services provided by the NFV-MANO. This is shown in chart A: Success of figure 4.1.3-1.

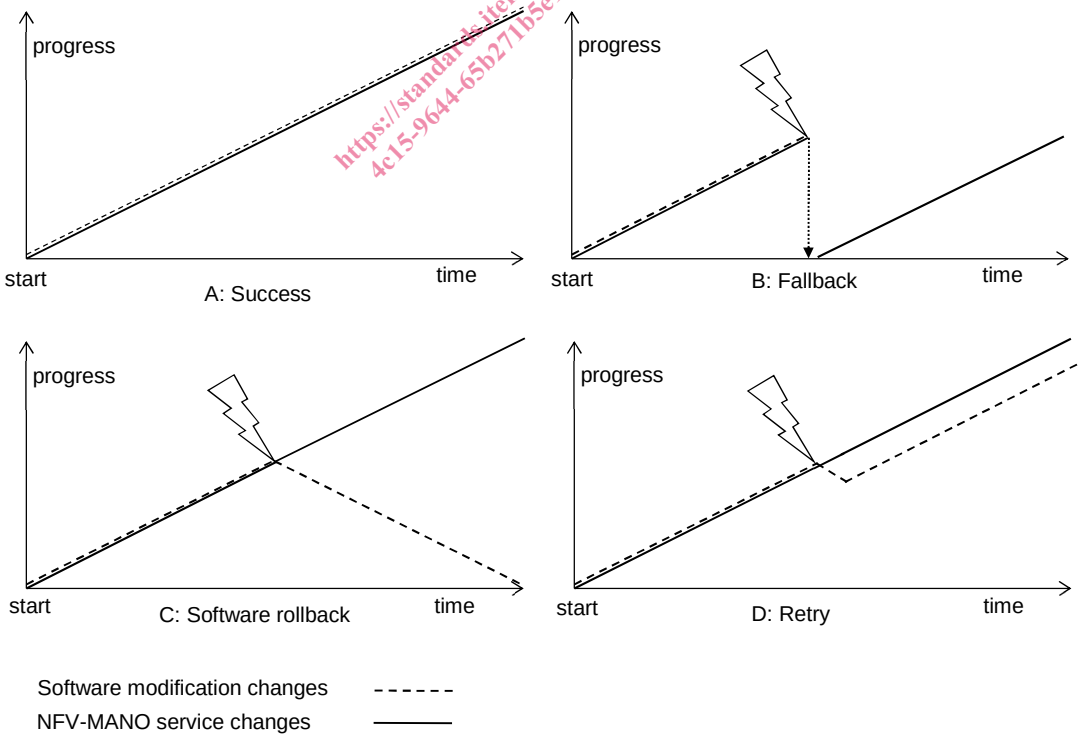

Figure 4.1.3-1: Fallback, rollback and retry of the software modification process