NFV Release 4 FEAT17 – CNF management concepts

Presented by: **Jörg Aelken**
FEAT17 feature prime,
IFA040 rapporteur

For: **ONAP**

2020-03-09

NFVIFA(20)000163

ETSI NFV Releases overview

Release 1

- Focus: the feasibility of NFV.
- Delivered the baseline studies and specifications.
- Set the NFV Architecture:
  - Infrastructure (NFVI),
  - Virtualized network functions (VNF),
  - Integration of the VNFs into Network Services (NS), and
  - NFV Management and Orchestration (NFV-MANO) aspects at different layers.

Release 2

- Focus: interoperability of NFV solutions.
- Details requirements and specification of interfaces and descriptors.
- Realizes the interoperability of solutions based on the NFV Architecture, detailing
  - VNF Package and VNF and NS Descriptors,
  - Acceleration,
  - Internal and external NFV-MANO interfaces.

Release 3

- Focus: feature enriching the NFV Architectural Framework, readying NFV for deployment and operation.
- Interfaces, modeling, etc. to support new features such as (not exhaustive list):
  - Policy framework,
  - VNF snapshot,
  - NFV-MANO management,
  - Multi-site,
  - Cloud-native, etc.

Release 4

- Focus: orchestration, cloudification and simplification of network deployment and operations.
- Interfaces, modeling, etc. to support new features such as (not exhaustive list):
  - Container-based deployments,
  - Further 5G support,
  - Service-based architecture concepts,
  - Generic OAM functions, etc.

Release 4: overview

Strawman of the Release 4 Definition available since NFV#26, which documents about new features for Release 4.

4 main technical areas and 14 features (7 new features, 5 carried-over from Release 3, and 2 enhancement features comprising “specific technical enhancements” and security aspects).

Technical areas

- NFVI evolution
- Enhancing NFV automation and capabilities
- Evolving NFV-MANO framework
- Operationalization

Features in the Release 4 definition

New features

- Network connectivity integration and operationalization
- Multi-tenancy enhancements for NFV-MANO
- NFV-MANO automation and autonomous networks
- SBA for NFV-MANO
- NFV enhancements for 5G
- VNF common management functions
- Continuous VNF integration

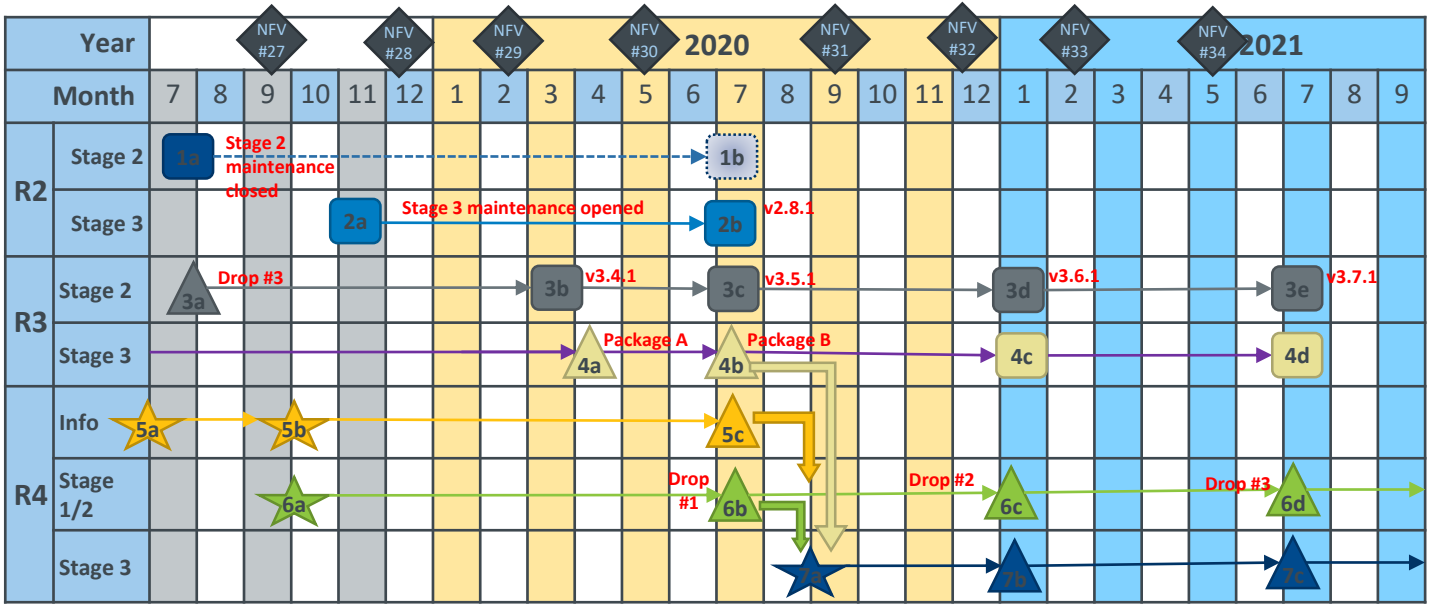
Enhancement features

- NFV security hardening
- Specific technical enhancements

Carried over from Rel. 3

- NFV-MANO upgrade
- MEC in NFV
- Licensing management
- Cloud-native VNFs and Container Infrastructure management
- Security management

ETSI ISG NFV release Schedule



NOTE: icon numbers meaning, refer to the slide's note section.

FEAT17

”Cloud-native VNFs and Container Infrastructure management”

Solution aggregated as NFV Release 4 feature

NFV Release 4 feature #17 "**Cloud-native VNFs and Container Infrastructure management**"

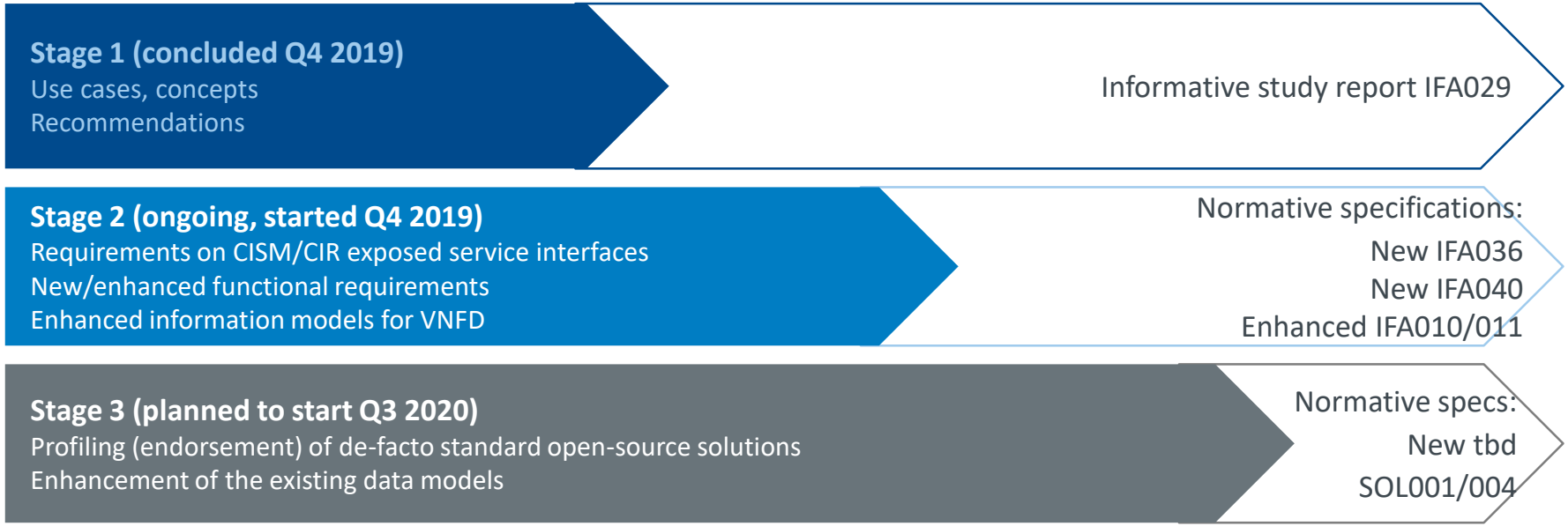
FEAT17 scope (according to NFV Release-4 definition):

- NFV Architecture support for VNFs which follow "cloud-native" design principles.
- Enhance NFV-MANO capabilities to support container technologies based on ETSI GR NFV-IFA 029.
- Enhance NFV-MANO capabilities for container management and orchestration
- Enhance information model for containerized VNFs both using bare metal or nested virtualization technologies

FEAT17 deliverables (new work items):

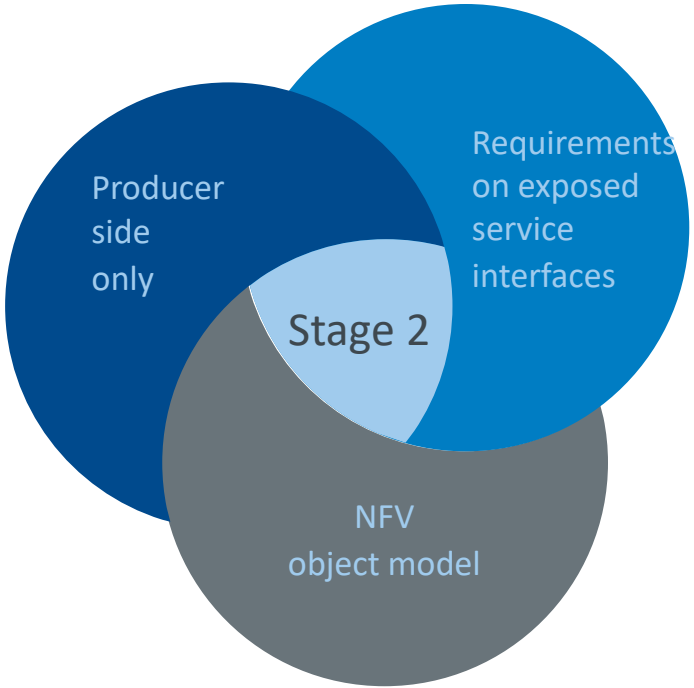
- ETSI GS NFV-IFA 036 – "Specification of requirements for the management and orchestration of container cluster nodes"
- ETSI GS NFV-IFA 040 – "Requirements for service interfaces for OS container management and orchestration"
- ETSI GS NFV-SEC 023 – "Container Security Specification"

FEAT17 implementation plan



Detailed feature implementation status at: [ETSI NFV public wiki]

New stage 2 approach for IFA036 and IFA040



Specify NFV object model

- Description/definition of the NFV object model for container management and orchestration in NFV-MANO.
- Provides abstract terminology to be used in requirement specifications.
- Enables mapping to de-facto standard solution objects in stage 3.

Specify requirements on exposed service interfaces

- Define services of specified functions and requirements on their exposed interfaces.
- Do **not** specify interface operations or information models
- Enable profiling of de-facto standard open source solutions as stage 3.

Specify producer side only

- No constraints on the consumers, no reference points with 1:1 consumer/producer relations.
- Enable flexible implementation of services, no mandated architectural option.

IFA029

“Report on the Enhancements of the NFV architecture towards "Cloud-native" and "PaaS"”

Overview IFA029

PaaS and general cloud native

Concepts for PaaS service types: VNF Common and Dedicated services

Use cases for PaaS-type capabilities

Potential architectural enhancements on PaaS related use cases:

- PaaS services are modelled as VNFs
- PaaS services are modelled as a new type of NFVI resources
- PaaS services are modelled as a new type of object specific to the PaaS layer

Recommendations:

- Specify a function for a PaaS Service Descriptor Catalogue
- Specify a function for a PaaS Service Registry

OS Containers

Use cases specific to utilization of containers

- Container platform deployment scenarios (e.g. nested or bare-metal)
- Containerized VNF LCM examples

Description of functions:

- Container Infrastructure Service (CIS)
- Container Infrastructure Service Management (CISM)
- Container Image Registry (CIR)

Potential architectural enhancements, including CISM to NFV-MANO mapping options

Comparison of architectural options, but no conclusion/recommendation

Recommendations for OS containers

- Specify MCIO and MCIOP
- Enhance the VNFD with references to MCIOs and MCIOPs
- Specify interfaces of Container Infrastructure Service Management (CISM) function
- Specify a function for Container Image Registry (CIR)

Published version available at: [ETSI GR NFV-IFA 029 V3.3.1 (2019-11)]

New term definitions (transferred to NFV003 official NFV terminology)

Term	Specification
Container Infrastructure Service (CIS)	Service that provides runtime environment for one or more container virtualisation technologies
Container Infrastructure Service instance	instance providing runtime execution environment for container
Container Infrastructure Service Management (CISM)	function that manages one or more Container Infrastructure Services
Container Image Registry (CIR)	function that stores container images and makes them available to other functions
Managed Container Infrastructure Object (MCIO)	object managed and exposed by the Container Infrastructure Service Management, representing the desired and actual state of a containerized workload, including its requested and allocated infrastructure resources and applicable policies
Managed Container Infrastructure Object Package (MCIOP)	aggregate of declarative descriptor and configuration files for multiple Managed Container Infrastructure Objects

Mapping of terms to de-facto standard solutions

Term	Analogy
Container Infrastructure Service (CIS)	Kubernetes services exposing CRI, CNI, CSI
Container Infrastructure Service (CIS) instance	Kubernetes worker node
Container Infrastructure Service Management (CISM)	Kubernetes master node & Helm 3 client
Managed Container Infrastructure Object (MCIO)	K8s managed objects (Pods, Deployments, Persistent Volume Claim, Service, etc.)
Managed Container Infrastructure Object Package (MCIOP)	Helm charts

IFA040

"Requirements for service interfaces for OS container management and orchestration"

Overview IFA040

Overview and framework for OS container management and orchestration

Formal, normative specification of new functions:

- Container Infrastructure Service Management (CISM)
- Container Image Registry (CIR)

Formal specification of service and service interface concepts for the new functions, anchoring the new approach for stage 2

OS container NFV object model

Formal, normative specification of managed objects and their relationship to existing NFV information model:

- Managed Container Infrastructure Object (MCIO)
- Managed Container Infrastructure Object Package (MCIOP)
- Namespace
- Namespace quota
- OS Container Image

Specification of CISM services and requirements on their interfaces

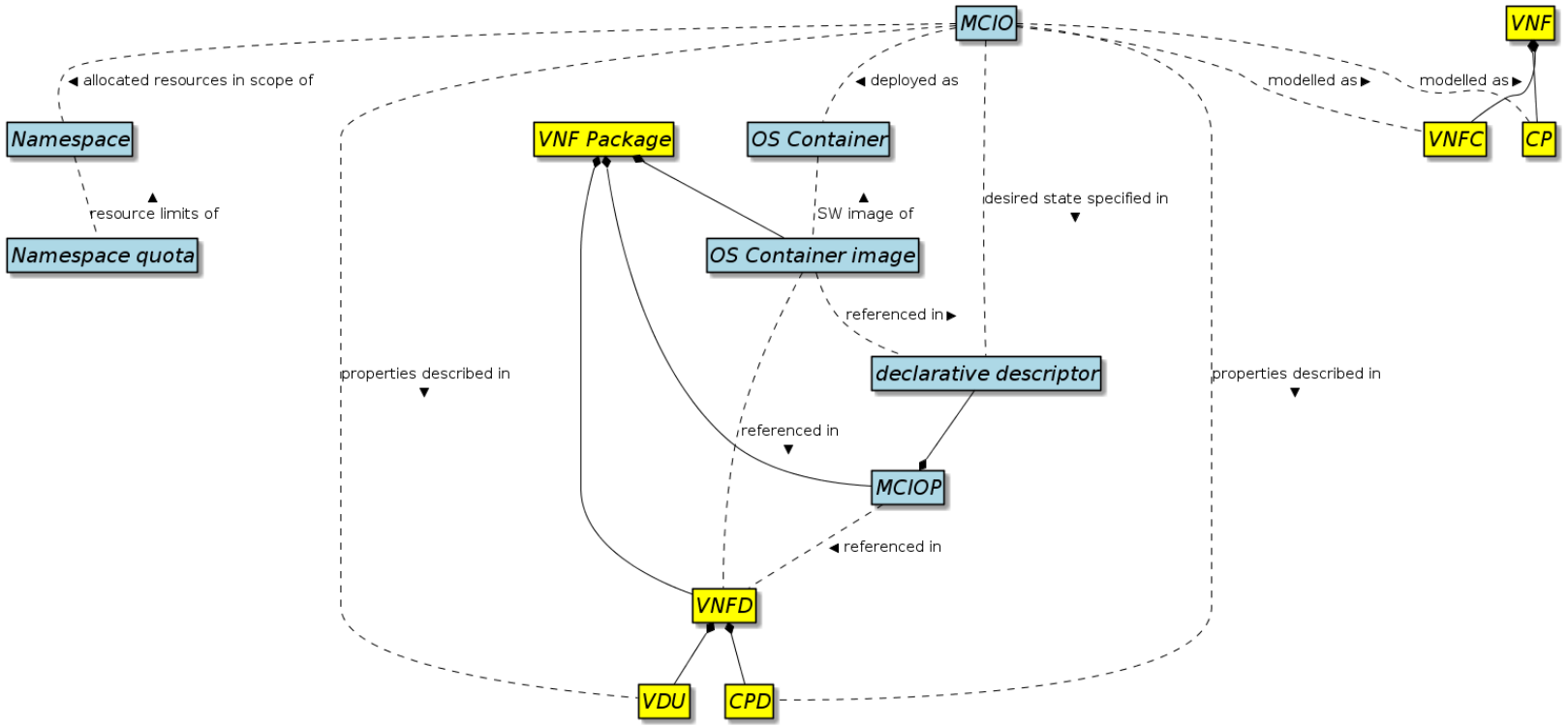
- OS container workload management service
- OS container compute management service
- OS container storage management service
- OS container network management service
- OS container configuration management service

Specification of CIR service and requirements on its interface

- OS container image management service

Latest draft available at: [NFV open draft area IFA040]

NFV OS Container objects relationship



Mapping to de-facto standard solutions

IFA040 service interface	De-facto standard profiling target
OS container workload management service	Helm 3 API
OS container compute management service	Kubernetes native workloads APIs
OS container storage management service	Kubernetes native storage APIs
OS container network management service	Kubernetes native service APIs
OS container configuration management service	Kubernetes native configuration and parts of cluster APIs
OS container image management service	Docker Registry API

Enhanced IFA010

"Functional requirements specification"

Agreed new functional requirements for CISM and CIR

CISM functional requirements for

- OS container infrastructure resource management
- MCIO management
- management of containerized workloads based on MCIOPs
- OS container configuration management
- OS container image management

CIR functional requirements for

- OS container image management

The CISM function is exposing OS container management service interfaces on different abstraction levels:

- The “OS container workload management service” exposes a management service interface on MCIOP abstraction level.
- The “OS container compute/storage/network management services” expose management service interfaces on MCIO abstraction level.

Enhanced IFA011

“VNF Descriptor and Packaging Specification”

Agreed VNFD information model changes

New requirements for the description of VNF Package content

- The VNF Package shall contain one or more MCIOPs
- The VNFD shall support the possibility to reference one or more MCIOP(s)
- The VNFD shall support the possibility to reference OS container images

Enhanced VNFD information model

- Introduce new IE for OsContainerDesc
- Enhance the VDU IE to model a MCIO as VDU (model a K8s Pod as VDU)
  - Add attribute for OsContainerDesc
- Enhance the VNFD IE with an attribute, referencing included MCIOPs
- Allow hybrid VNFs, i.e. VM-based and OS container based VNFCs
- Forbid hybrid VNFCs, they have to be either VM- or OS container based
- Enhanced IE for SwImageDesc to reflect capabilities for OS container images