

Open Infrastructure Summit 2019 Shanghai

NTT DOCOMO's Operational Challenges of Commercial Multi-vendor NFV System

Hayashi Kohei, NTT DOCOMO



Jo Hiroyuki, NTT

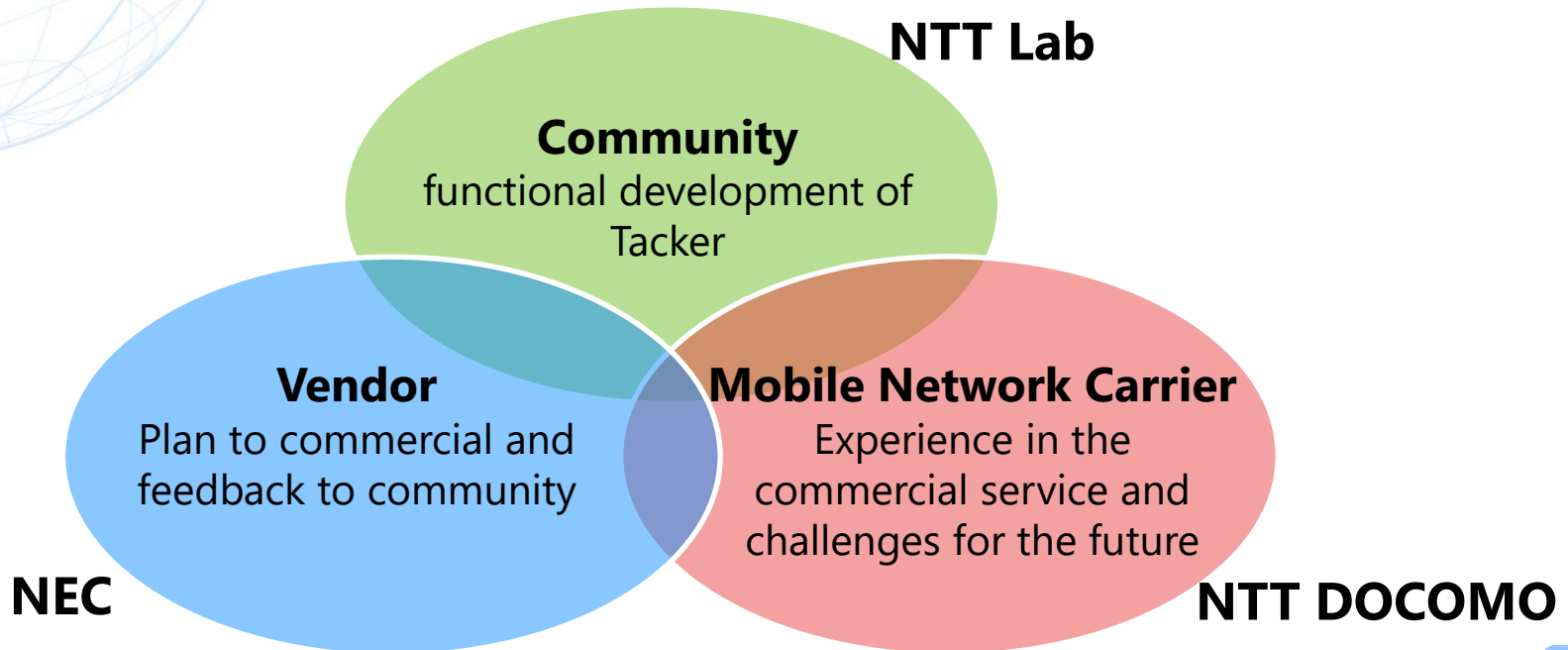


Takahashi Toshiaki, NEC Corporation



Message of this session

- Technical challenges that DOCOMO are experiencing in operating NFV in commercial services
- Solution for the challenges with the Open Infrastructure Community



About us

Hayashi Kohei

- NTT DOCOMO
- Developing a mobile core network in telecom operator
- System architect of NFV MANO



Takahashi Toshiaki

- NEC Corporation
- Providing a virtualization infrastructure for telecom operators
- Joining Tacker community



Jo Hiroyuki

- NTT Network Systems Laboratories
- OpenStack Tacker core
- NFV MANO system developer
- 10+ years system engineer in Telco industry

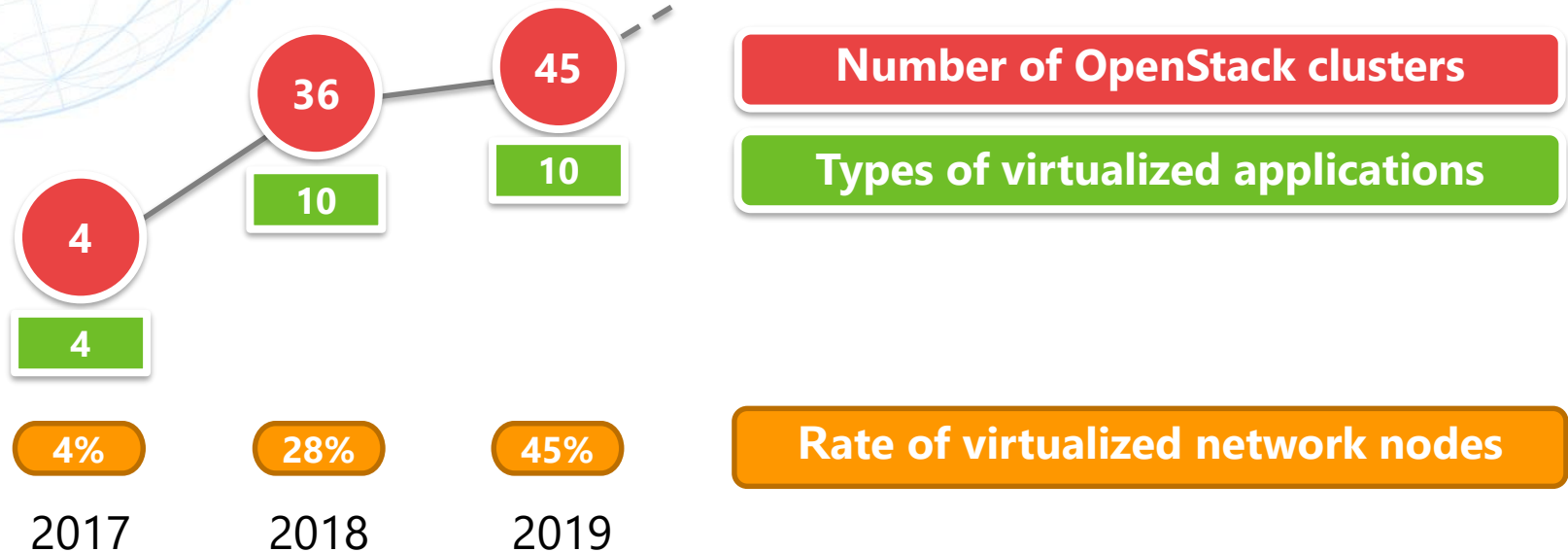




1. Current Status and Technical Challenges of NFV

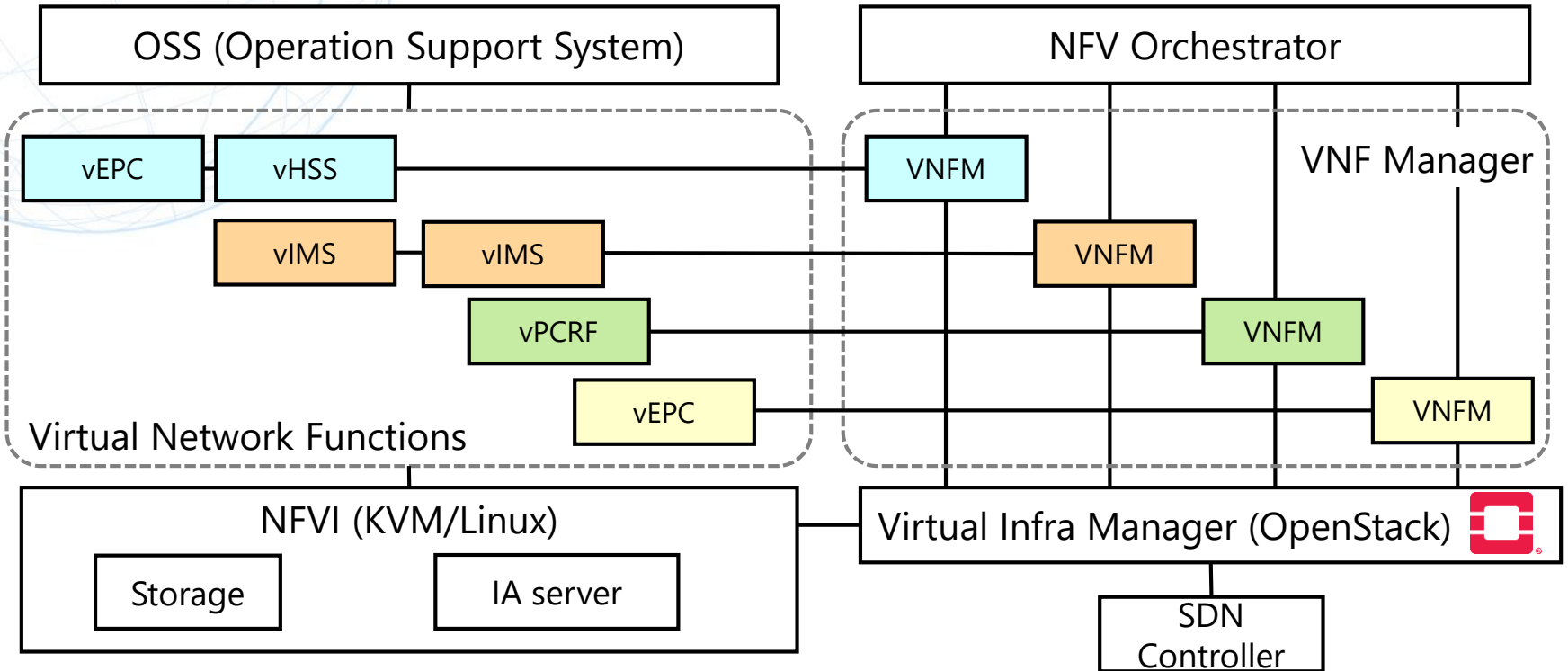
Scale of commercial NFV deployment

- DOCOMO had started operations of vEPC system on commercial network in 2016
- 45% of network nodes in core network are virtualized



DOCOMO NFV configuration

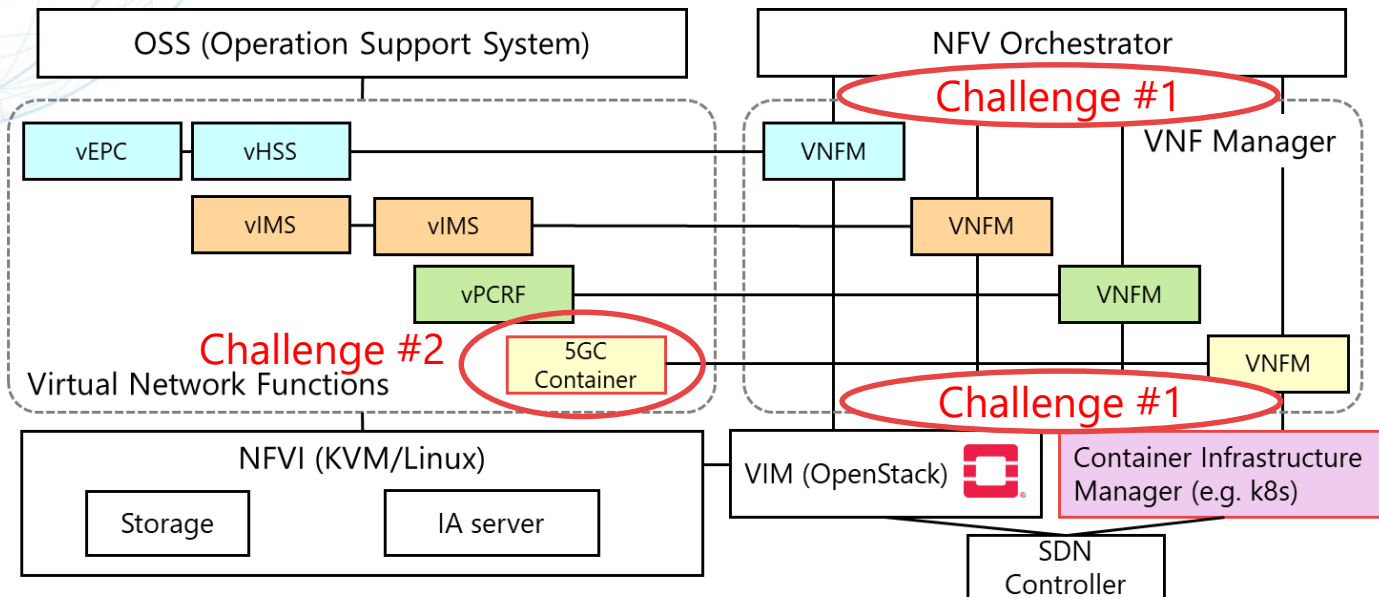
Multi-function/multi-vendor ETSI NFV-based architecture on a unified OpenStack infrastructure



Overview of technical challenges



2 technical challenges mapped in the DOCOMO NFV configuration

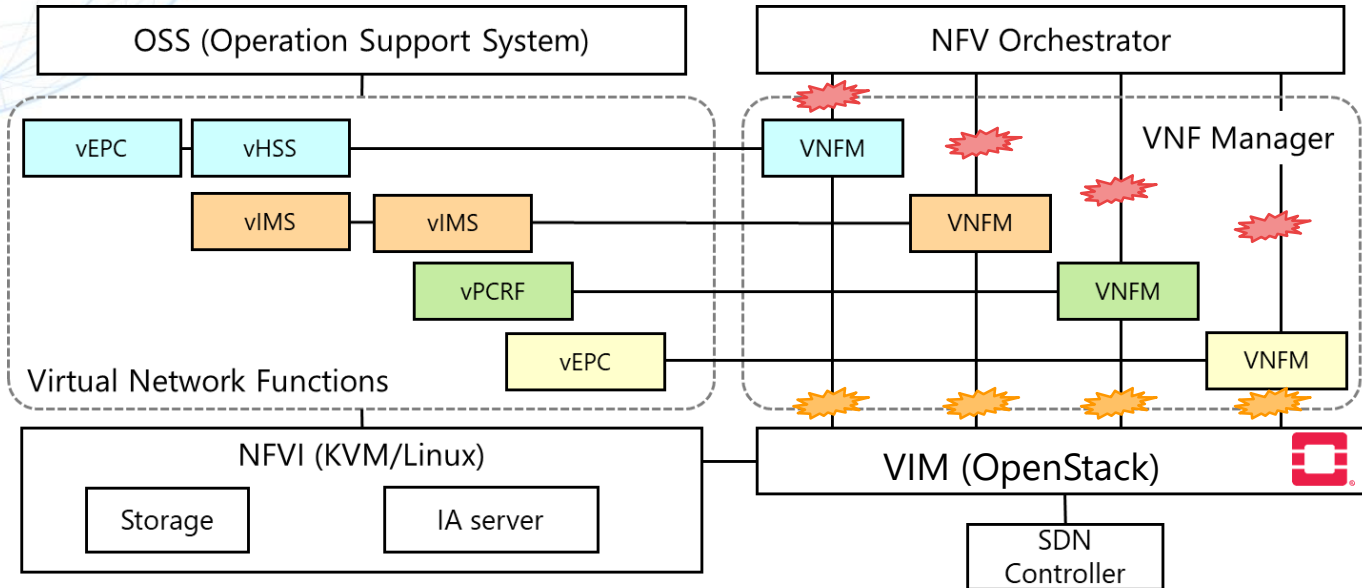
- Challenge #1: mitigation of many interfaces between specific VNFM for each VNF vendor and other components
- Challenge #2: support for 5G Core and handling of containerized VNF by NFV-MANO



Challenge #1 : Mitigation of integration and verification in NFV-MANO and OpenStack

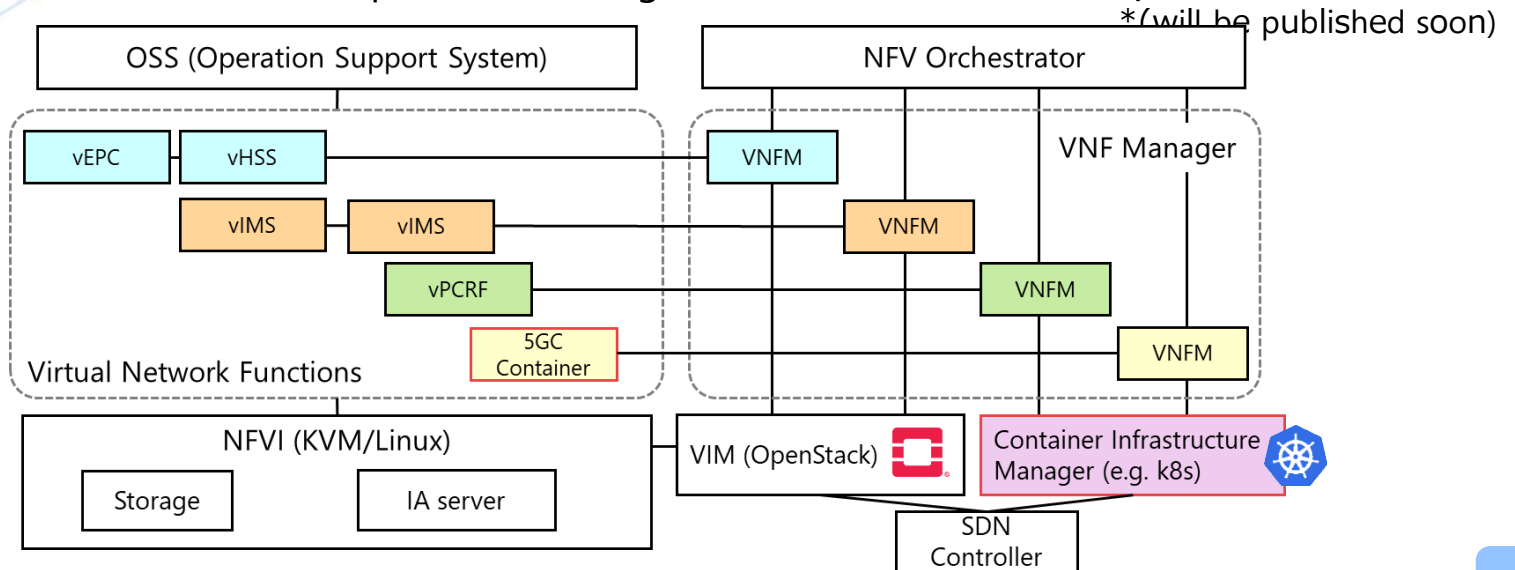
Integration and verification increases with upgrades of VNFM, NFVO, VIM

- Integration and verification between VNFM and NFVO 
- Compatibility validation between VNFM and VIM for the upgrading of OpenStack API 



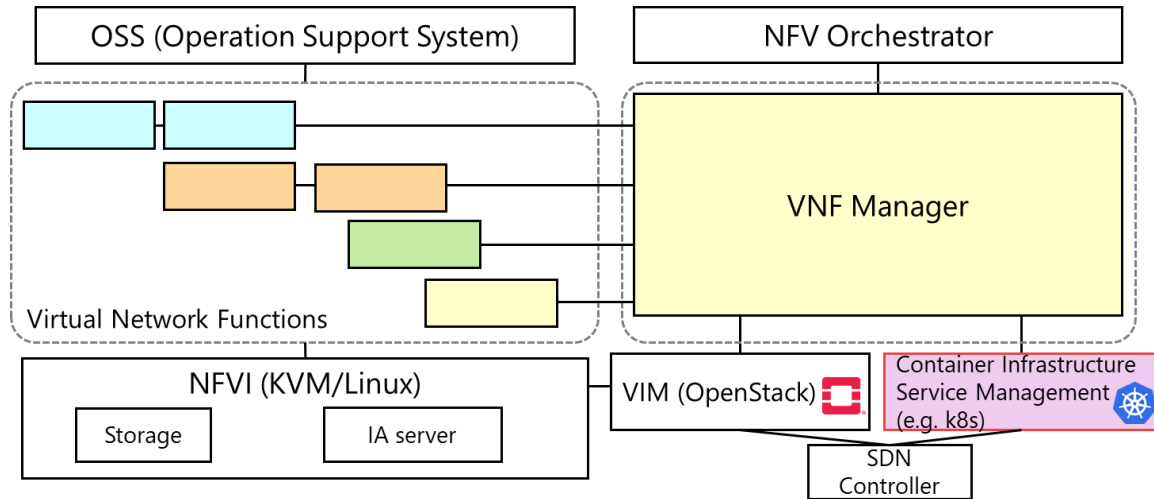
Challenge #2 : Support 5G Core and handling of containerized VNF by MANO

- Since 5GC may introduce container-based VNFs, NFV-MANO should support the component of Container Infrastructure Service Management (CISM) such as Kubernetes
- Requires support for co-existence of container-based and VM-based VNFs
- Different options to introduce containers to NFV
 - ETSI NFV discussed several options to manage containerized VNFs in report IFA029*



NFV MANO for the future core network

- Next generation VNFM that is able to solve technical challenges
 - Standard compliant implementation and OpenStack API support by VNFM leads to reduced costs of integration and verification with NFVO, VIM and CISM
- We will accelerate the OpenStack Tacker as the open source VNFM that supports various types of VNF and Virtualization Infrastructure (e.g. OpenStack and Kubernetes)
 - Designing the Tacker based on experience developing and operating multi-vendor NFVs
 - Feedback to ETSI NFV specification based on development of Tacker

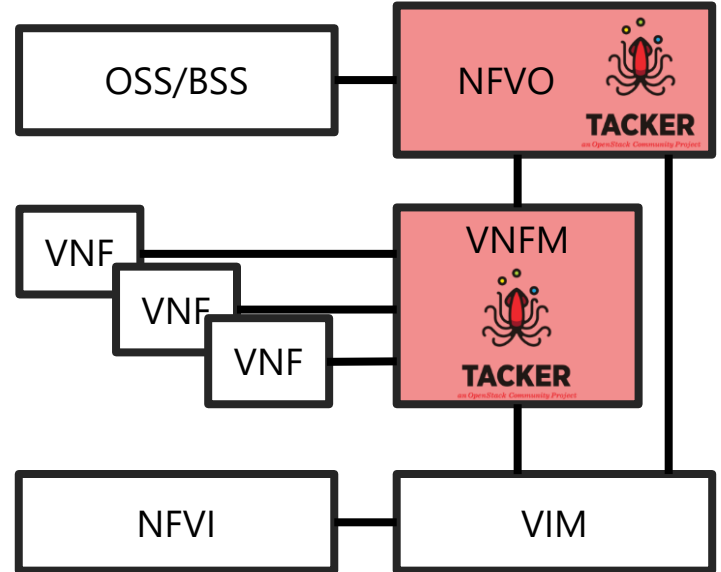




2. OpenStack Tacker as Next Generation VNFM

OpenStack Tacker

- Official OpenStack project
- Aiming at implementing VNFM and NFVO
- Orchestrating virtualised telecom infrastructure
 - physical and virtual infrastructure
 - virtualised network and applications



Why Tacker?

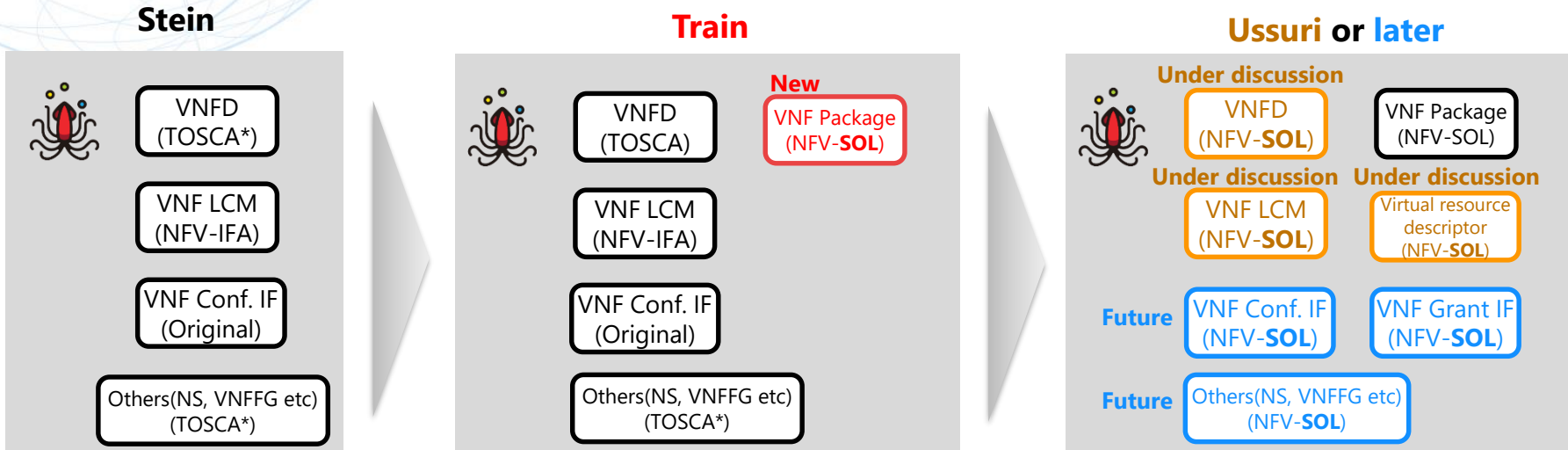
- For Challenge #1: ETSI NFV compliant VNFM
 - NFV-SOL compliant API enables mitigation of many interfaces between VNFM and other component
 - <https://www.etsi.org/standards-search#&search=NFV-SOL>
 - Explained in "3. Tacker Function Enhancement ~VNF Lifecycle Management~"
- For Challenge #2: Containerized VNF support to Kubernetes VIM
 - Implemented in Queens release
 - Need discussion for real use case
 - Explained in "3. Tacker Function Enhancement ~ Container Support ~ "



3. Tacker Function Enhancement ~ VNF Lifecycle Management ~

NTT is Enhancing Tacker VNF LCM

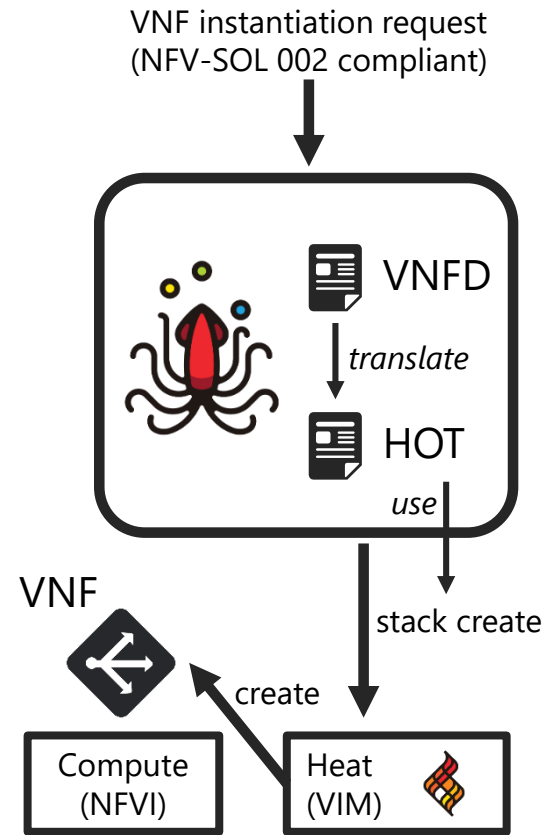
- Before Stein, Tacker referred to ETSI **NFV-IFA** standard, which is a functional specification rather than an API specification.
- ETSI **NFV-SOL** API Specification was published in 2018. NTT started to propose and implement **SOL** compliant API.



*) TOSCA Simple Profile in YAML and TOSCA Simple Profile for NFV
<http://docs.oasis-open.org/tosca/TOSCA-Simple-Profile-YAML/v1.1.1/TOSCA-Simple-Profile-YAML-v1.1.html>
<http://docs.oasis-open.org/tosca/tosca-nfv/v1.0/tosca-nfv-v1.0.html>

How to VNF LCM? – translation of VNFD to HOT

- When LCM, VNFD is translated to HOT (Heat Orchestration Template).
 - As Tacker utilizes Heat when creating/scaling/healing /deleting virtual resources as a component of VIM.
- 2 translation procedures are planned to be implemented.
 - Static translation (legacy approach)
 - Descriptor-based virtualised resource management
 - considering VIM configuration into translation logic without bringing VIM specific stuff into VNFM
- Operators can choose the procedure to use by specifying "*additionalParams*" in VNF LCM request.



Static translation

- Basic VNFD types including VDU, BlockStorage, CP, VL, ScalingAspects, InstantiationLevels will be supported.

SOL 001 VNFD type	HOT resource type
tosca.nodes.nfv.Vdu.Compute	OS::Nova::Server
tosca.nodes.nfv.Vdu.VirtualBlockStorage	OS::Cinder::Volume
tosca.nodes.nfv.VduCp	OS::Neutron::Port
tosca.nodes.nfv.VL	OS::Neutron::Net, OS::Neutron::Subnet OS::Neutron::QoSBandwidthLimitRule OS::Neutron::QoSPolicy
tosca.policies.nfv.ScopingAspects tosca.policies.nfv.VduInitialDelta tosca.policies.nfv.VduScopingAspectDeltas	OS::Heat::AutoScalingGroup OS::Heat::ScopingPolicy

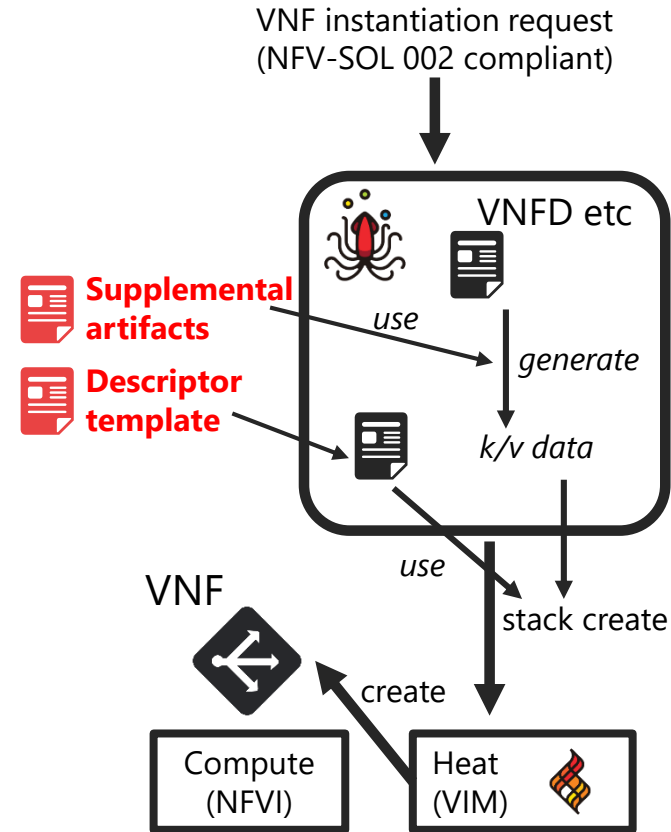
- Limitation due to gaps between VNFD and HOT.
 - If no ScopingAspect is defined for a VDU, number of the VDU is always 1 (i.e. VduInstantiationLevels is ignored).
 - Only one ScopingAspectDeltas per ScopingAspect is valid.

For commercial use cases...

- In a real complex use case, VIM and NFVI configuration and availability must be considered.
 - A operator must design CPU pinning assignments at the physical level.
 - Depending on the VNF, a operator may want to attach external storage instead of Cinder.
 - A operator can modify the high availability design of VNF depending on the availability of VIM/NFVI.
- Current SOL 001 doesn't allow to describe above attributes in VNFD.
- Static translation is applicable to limited use cases.

Descriptor-based virtualised resource management*

- SOL 014 is now making effort to specify the data model of virtual resource descriptor template.
- Key aspects of this approach are:
 - descriptor template is not depending on VIM configuration.
 - key-value parameters are generated from VNFD etc. using supplemental artifacts provided per VNF and VIM.
 - The choice of value is made under consideration of VIM configuration.
- NTT is planning to implement this approach but SOL 014 is still in draft and we expect it will be published in near future.



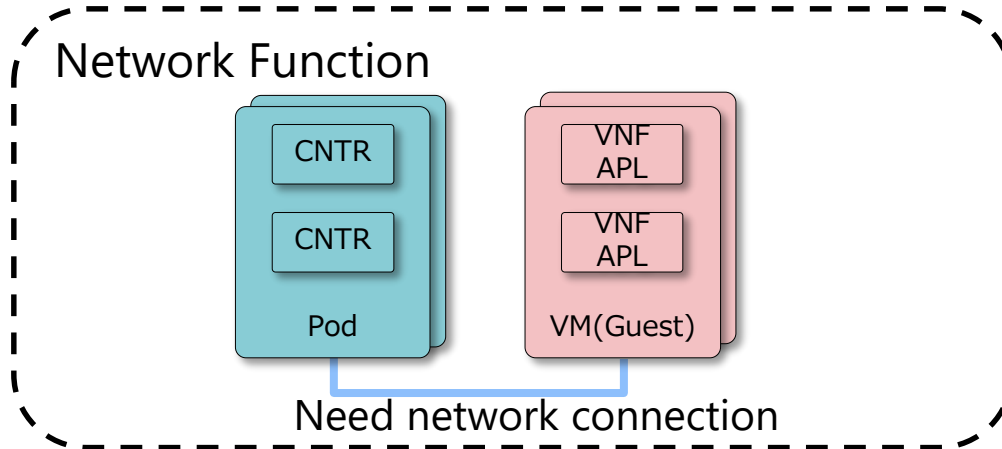


3. Tacker Function Enhancement

~ Container Support ~

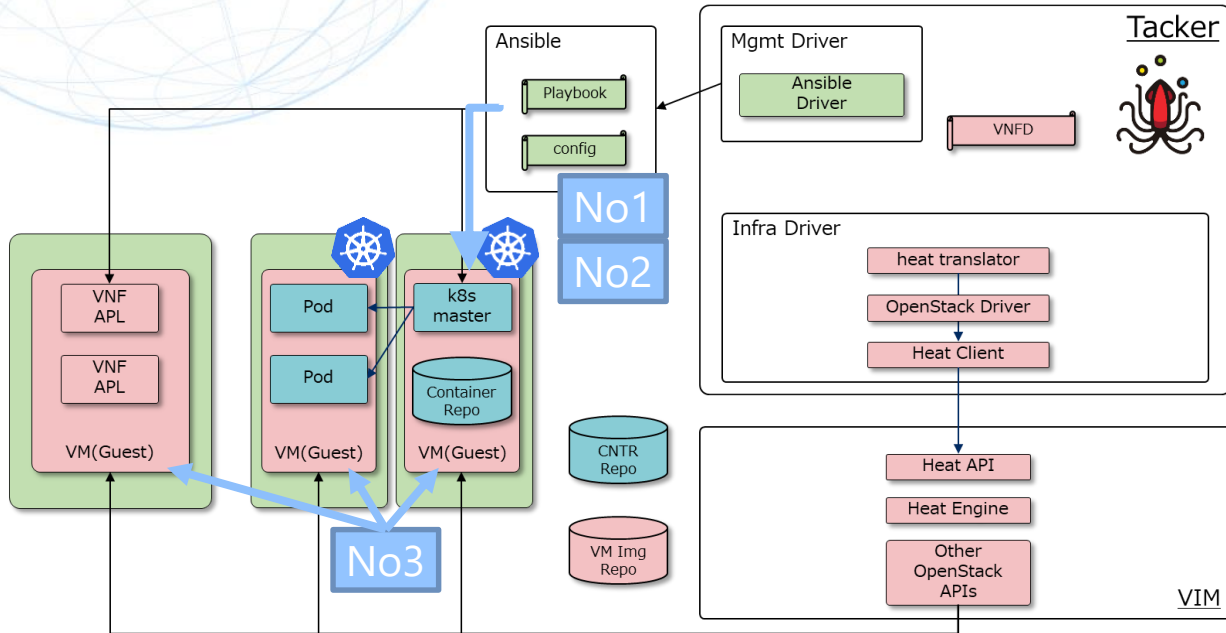
Enhancement of container support (k8s)

- Tacker already supports Kubernetes VIM
- What should we do more? Network Functions consist of containers and virtual machines.
 - Deploy containers and virtual machines in single operation from Tacker
 - Connect between containers and virtual machines
- Standardization: ETSI GR NFV-IFA 029 V0.20.0 (2019-08) * Final draft
 - https://docbox.etsi.org/ISG/NFV/Open/Drafts/IFA029ed311_Arch_enhancement_for_Cloud-native_&_PaaS/NFV-IFA029v0200.docx
 - Document shows different **Kubernetes to NFV-MANO mapping options**
 - We have implemented options #6 (implementation 1) and #3 (implementation 2) as described in the next slides.



Implementation 1

- Option#6: CISM* embedded into VNF without support for shared container service (* CISM=Kubernetes)
 - Kubernetes functionality is embedded into a VNF and uses the NFVI resources allocated to the VNF.



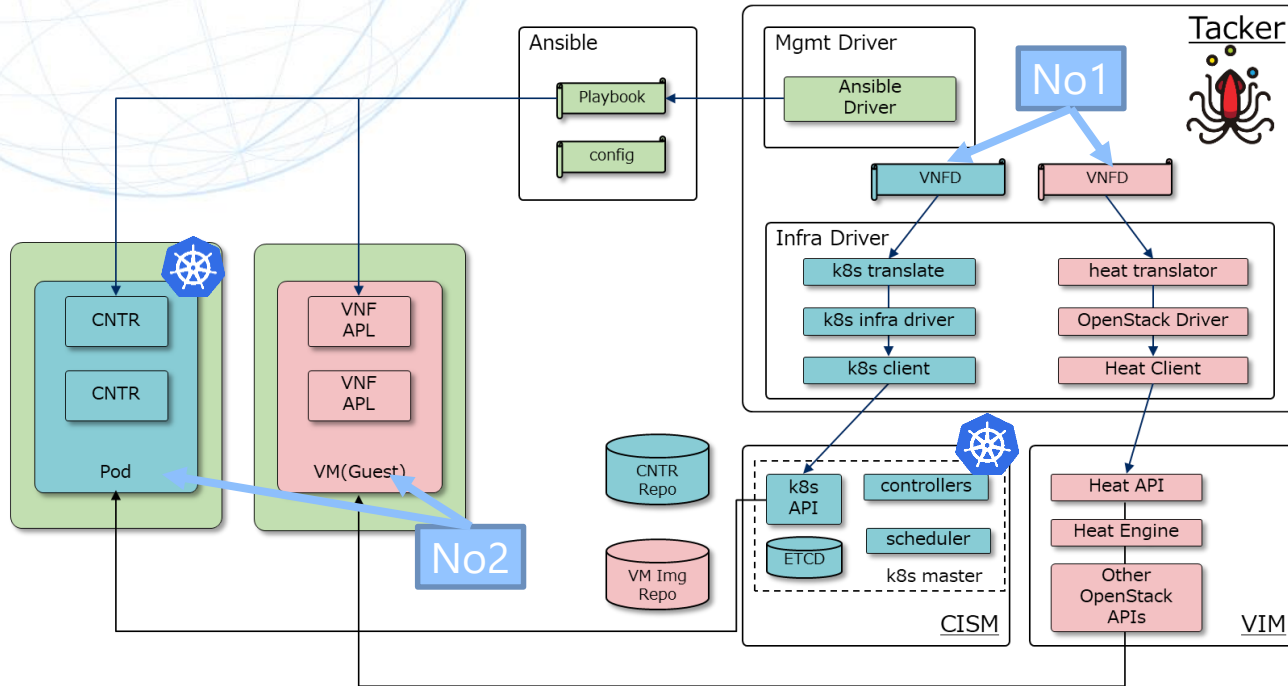
No	Added item
1	Kubernetes cluster configuration playbook
2	Pods creation playbook
3	Network connection setup script

Implementation 1

- Knowledge
 - We can implement this.
 - Kubernetes master/worker nodes are created using Kubernetes node images.
 - Kubernetes configuration is executed using management driver.
 - Kubernetes is an application for Tacker.
- Future challenges
 - We need to design the detailed network between containers and VM.
 - Container Life cycle is hidden and cannot managed from Tacker.
 - E.g. Tacker cannot catch container restart events.

Implementation 2

- Option#3: CISM* as a stand-alone functional block (* CISM=Kubernetes)
 - Kubernetes functionality is assigned to new NFV-MANO functional block.



No	Added item
1	Multiple VIM management feature
2	Network connection setup script

Implementation 2

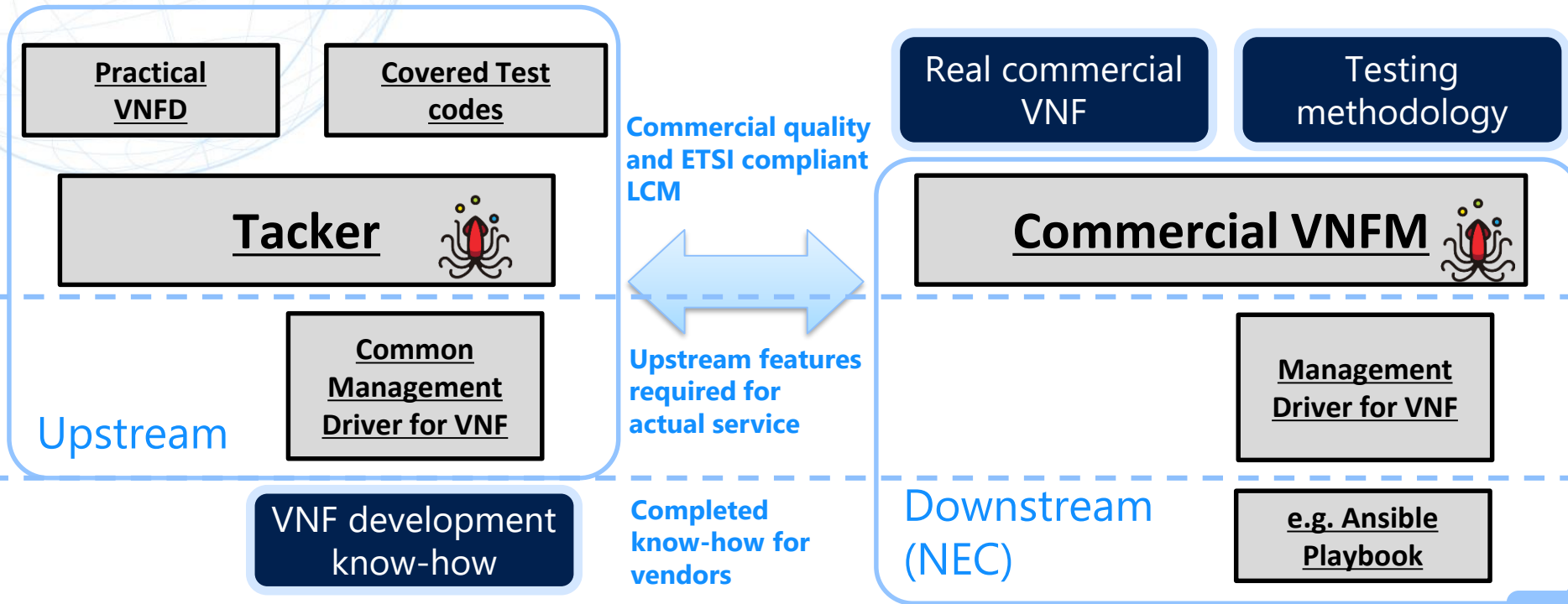
- Knowledge
 - We can implement this.
 - Tacker can manage Kubernetes.
- Future challenges
 - We need to design the detailed network between containers and VM.
 - How to connect between different infrastructures
 - We need to refine features. We will discuss it in Tacker community.
 - Tacker support capabilities expansion
 - Multiple VIM management
 - Not VNFM, how to implement?
 - VM and Container VNFD operation



4. Tacker Quality Enhancement for Commercial Service

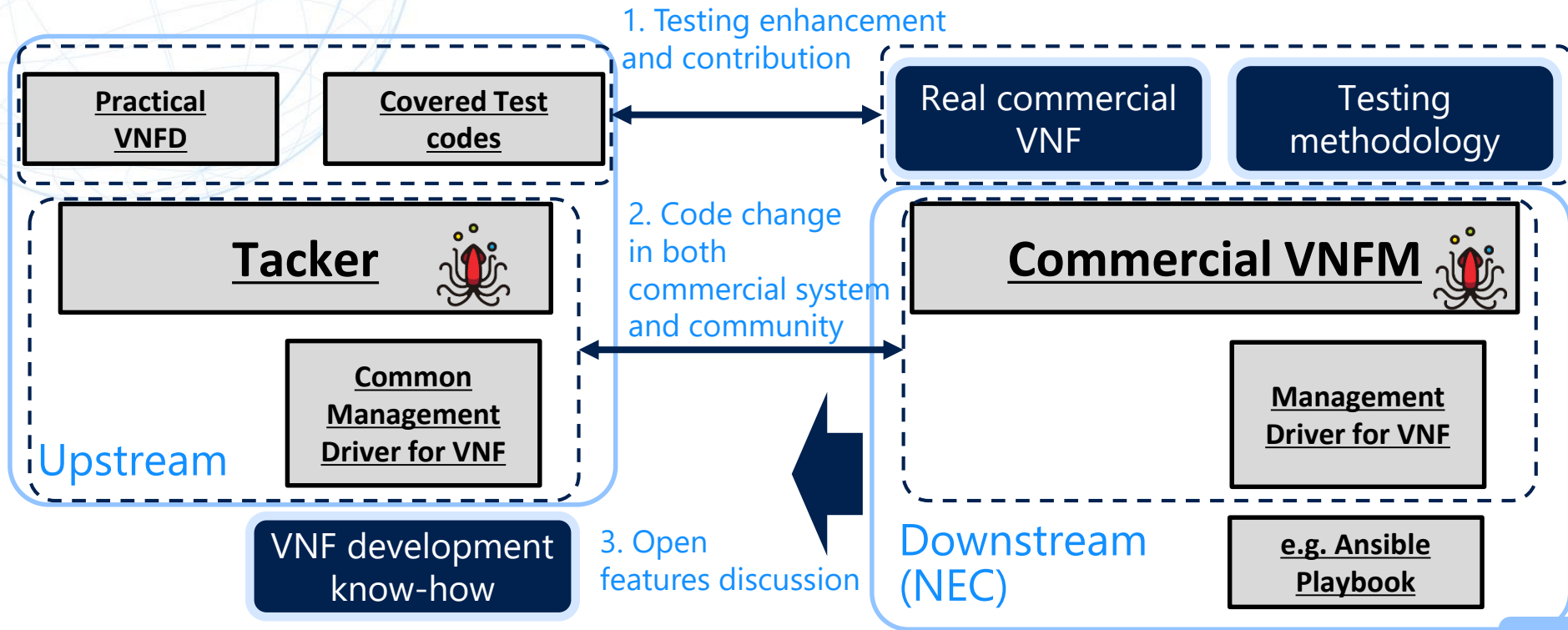
Quality enhancement for commercial service

- NEC is proceeding quality enhancement activities for commercial service.
- NEC plans to contribute our activities to make Tacker a high quality VNFM.



Quality enhancement for commercial service

- NEC is proceeding 3 activities for Tacker quality enhancement.



Quality enhancement: Testing enhancement

Test types	Community	NEC
Unit tests	✓ Based on OpenStack guide	✓ Same level as community
Functional tests	✓ Integration tests with for each function e.g. <ul style="list-style-type: none"> • Create and delete VNF <ul style="list-style-type: none"> • Simple VNFD • Block storage attach • Multiple VDU • Placement (Affinity etc...) • Scaling • Monitoring/Alarming and action <ul style="list-style-type: none"> • Respawn • VDU-autohealing 	✓ Same level as community ✓ Combination tests based on real use case e.g. <ul style="list-style-type: none"> • Set scaling to VNF using block storage • Set scaling and autohealing <div data-bbox="1116 729 1862 852" style="background-color: #0070C0; color: white; padding: 10px; border-radius: 10px; text-align: center;"> Possible to test VNFM appropriately and systematically </div>
Abnormal system tests	✗ Limited to possible tests	✓ Enough manual & automatic tests <div data-bbox="1136 976 1843 1035" style="background-color: #0070C0; color: white; padding: 10px; border-radius: 10px; text-align: center;"> Possible to avoid fatal errors </div>

Quality enhancement: Testing enhancement

- NEC's focus

1. Appropriately covered test items

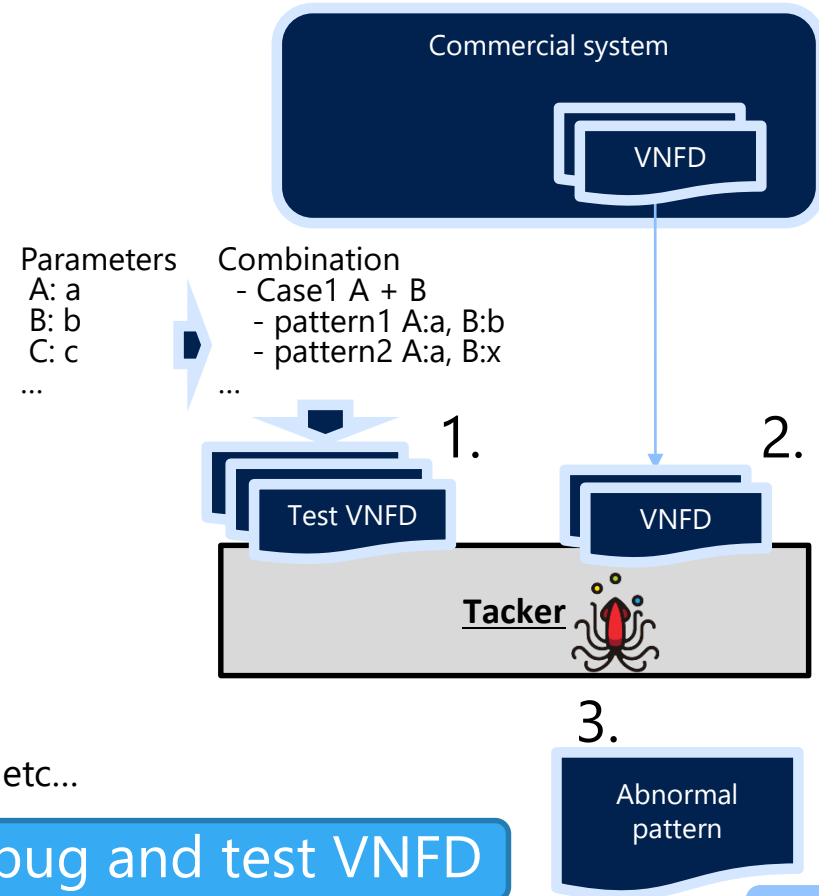
- Test VNFD parameter combinations
- Based on matrix of parameters used in commercial system

2. Commercial real use cases

- Test complicated VNF with VNFD used in commercial system

3. Abnormal system test

- Test abnormal system
 - e.g. DB becomes abnormal during LCM
- Detect the system is completely broken
 - e.g. inoperable situation, data corruption, etc...



Planning to contribute fix for detected bug and test VNFD

Quality enhancement: Code Change

- Fix Tacker bugs / Add missing features
 - Fix/Add Tacker codes in our commercial system if necessary
 - Contribute the change to Tacker community

Our contributions to Tacker community

Item	Status
Worked on fixing CLI issues and improvement	Fixed
Identifying & fixing gaps in scaling policies with volume bugs	Reported
Identifying & fixing gaps with mgmt-driver in different scenarios	Under review

Quality enhancement: Features discussion

- Know-how for VNF developers
 - Management driver development
 - Tacker provides the basement of management driver for VNF
 - We will open our know-how and discuss our and other vendor's know-how.
 - We have a plan to contribute common functions based on the discussion.
- Need to discuss real use cases
 - e.g. Scaling use cases
 - What do we want to scale in actual use cases?
 - Scale VDU? Scale 2 VDU together?
 - Current Tacker does not suit some use cases.
 - Tacker cannot heal VDU in scaling group.

Planning to open know-how and our internal discussions.

Summary

- Presented DOCOMO's achievements and forthcoming challenges
- Tacker as a solution for network carrier's large-scale NFV
- NEC and NTT, as Tacker community members, focuses on commercial use, K8s support and VNF LCM compliant with ETSI NFV.

