International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

Y.2321

(09/2016)

SERIES Y: GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

Next Generation Networks – Enhancements to NGN

Functional architecture for supporting virtualization of control network entities in next generation network evolution

Recommendation ITU-T Y.2321

ITU-T Y-SERIES RECOMMENDATIONS

GLOBAL INFORMATION INFRASTRUCTURE, INTERNET PROTOCOL ASPECTS AND NEXT-GENERATION NETWORKS, INTERNET OF THINGS AND SMART CITIES

GLOBAL INFORMATION INFRASTRUCTURE	
General	Y.100–Y.199
Services, applications and middleware	Y.200–Y.299
Network aspects	Y.300–Y.399
Interfaces and protocols	Y.400–Y.499
Numbering, addressing and naming	Y.500–Y.599
Operation, administration and maintenance	Y.600–Y.699
Security	Y.700–Y.799
Performances	Y.800–Y.899
INTERNET PROTOCOL ASPECTS	
General	Y.1000–Y.1099
Services and applications	Y.1100–Y.1199
Architecture, access, network capabilities and resource management	Y.1200–Y.1299
Transport	Y.1300–Y.1399
Interworking	Y.1400–Y.1499
Quality of service and network performance	Y.1500–Y.1599
Signalling	Y.1600–Y.1699
Operation, administration and maintenance	Y.1700–Y.1799
Charging	Y.1800–Y.1899
IPTV over NGN	Y.1900–Y.1999
NEXT GENERATION NETWORKS	
Frameworks and functional architecture models	Y.2000–Y.2099
Quality of Service and performance	Y.2100–Y.2199
Service aspects: Service capabilities and service architecture	Y.2200–Y.2249
Service aspects: Interoperability of services and networks in NGN	Y.2250–Y.2299
Enhancements to NGN	**Y.2300–Y.2399**
Network management	Y.2400–Y.2499
Network control architectures and protocols	Y.2500–Y.2599
Packet-based Networks	Y.2600–Y.2699
Security	Y.2700–Y.2799
Generalized mobility	Y.2800–Y.2899
Carrier grade open environment	Y.2900–Y.2999
FUTURE NETWORKS	Y.3000–Y.3499
CLOUD COMPUTING	Y.3500–Y.3999
INTERNET OF THINGS AND SMART CITIES AND COMMUNITIES	
General	Y.4000–Y.4049
Definitions and terminologies	Y.4050–Y.4099
Requirements and use cases	Y.4100–Y.4249
Infrastructure, connectivity and networks	Y.4250–Y.4399
Frameworks, architectures and protocols	Y.4400–Y.4549
Services, applications, computation and data processing	Y.4550–Y.4699
Management, control and performance	Y.4700–Y.4799
Identification and security	Y.4800–Y.4899
Evaluation and assessment	Y.4900–Y.4999

For further details, please refer to the list of ITU-T Recommendations.

Recommendation ITU-T Y.2321

Functional architecture for supporting virtualization of control network entities in next generation network evolution

Summary

Recommendation ITU-T Y.2321 introduces virtualization technology into next generation networks (NGNs), especially control functions. It also describes the functional architecture, functional entities (FEs), reference points and information flow to support virtualization of the control network entities (VCN) for NGN evolution. The architecture design fulfils the requirements proposed in Recommendation ITU-T Y.2320.

History

Edition	Recommendation	Approval	Study Group	Unique ID*
1.0	ITU-T Y.2321	2016-09-29	13	11.1002/1000/13012

Keywords

Network function virtualization, NGN, orchestration, resource management, VCN.

* To access the Recommendation, type the URL http://handle.itu.int/ in the address field of your web browser, followed by the Recommendation's unique ID. For example, http://handle.itu.int/11.1002/1000/11830-en.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications, information and communication technologies (ICTs). The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure, e.g., interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU had not received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

Table of Contents

Recommendation ITU-T Y.2321

Functional architecture for supporting virtualization of control network entities in next generation network evolution

1 Scope

This Recommendation describes the virtualization of control network entities (VCN) functional architecture that provides a virtualized running environment for next generation network (NGN) control functional entities, such as service control functions (SCFs) and content distribution functions (CDFs). The functional architecture of VCN for NGN evolution includes detailed capabilities of functions, functional entities (FEs) and reference points.

Annex A provides information flows for VCN.

2 References

The following ITU-T Recommendations and other references contain provisions which, through reference in this text, constitute provisions of this Recommendation. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; users of this Recommendation are therefore encouraged to investigate the possibility of applying the most recent edition of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is regularly published. The reference to a document within this Recommendation does not give it, as a stand-alone document, the status of a Recommendation.

[ITU-T Y.2012] Recommendation ITU-T Y.2012 (2010), *Functional requirements and architecture of next generation networks.*

[ITU-T Y.2320] Recommendation ITU-T Y.2320 (2015), *Requirements for virtualization of control network entities in next generation network evolution.*

[ITU-T Y.2701] Recommendation ITU-T Y.2701 (2007), *Security requirements for NGN release 1.*

3 Definitions

3.1 Terms defined elsewhere

This Recommendation uses the following terms defined elsewhere:

3.1.1 live migration [ITU-T Y.2320]: The process of moving a running virtual machine (VM) between different physical machines without interrupting the operating system and applications in the virtual machine.

3.1.2 virtualized control network entity (VCNE) [ITU-T Y.2320]: A control network entity deployed on virtualized infrastructure (i.e., across one or multiple virtual machines (VMs)).

3.1.3 virtualized control network entity descriptor (VCNED) [ITU-T Y.2320]: A configuration template that describes a virtualized control network entity (VCNE) in terms of its deployment and operational behaviour, and that is used in the management of the lifecycle of a VCNE instance.

3.1.4 virtual machine (VM) [ITU-T Y.2320]: The virtualized computation environment that behaves like a physical server.

3.2 Terms defined in this Recommendation

None.

4 Abbreviations and acronyms

CDF Content Distribution Function

CPU Central Processing Unit

CPCI Compact Peripheral Component Interconnect

FE Functional Entity

IP Internet Protocol

MF Management Function

MMCF Mobility Management and Control Function

NGN Next Generation Network

NIC Network Interface Card

NS Network Service

NSD Network Service Descriptor

PNF Physical Network Function

QoS Quality of Service

SCF Service Control Function

VCN Virtualization of Control Network entities

VCNE Virtualized Control Network Entity

VCNED Virtualized Control Network Entity Descriptor

VCNM VCN Manager

VI Virtualized Infrastructure

VIM Virtualized Infrastructure Manager

VLAN Virtual Local Area Network

VLD Virtual Link Descriptor

VM Virtual Machine

5 Conventions

In this Recommendation:

The phrase "is required to" indicates a requirement that must be strictly followed and from which no deviation is permitted if conformance to this Recommendation is to be claimed.

The phrase "is prohibited from" indicates a prohibition that must be strictly followed and from which no deviation is permitted if conformance to this Recommendation is to be claimed.

The phrase "is recommended" indicates a requirement that is recommended, but which is not absolutely required. Thus, this requirement need not be satisfied to claim conformance.

The phrase "is not recommended" indicates a requirement that is not recommended, but which is not specifically prohibited. Thus, conformance with this Recommendation can still be claimed even if this requirement is satisfied.

The phrase "can optionally" indicates an optional requirement that is permissible, without implying any sense of being recommended. This term is not intended to imply that the vendor's implementation must provide the option and the feature can be optionally enabled by the network

operator/service provider. Rather, it means the vendor may optionally provide the feature and still claim conformance with the Recommendation.

6 Overview

The hardware of telecommunication networks is changing from dedicated compact peripheral component interconnect (CPCI) to general purpose servers and virtualization technology is being applied to telecommunication networks. These new trends bring great benefits, such as rapid deployment and updating, energy saving and hardware sharing.

Traditionally, NGNs [ITU-T Y.2012] have defined architecture, capability of different functions/functional entities (FEs) and reference points between those functions/FEs. However, new trends affect the infrastructure in ways that are not specified in [ITU-T Y.2012].

Therefore, to achieve such benefits, NGN architecture needs to be improved to meet virtualization-related requirements [ITU-T Y.2320], e.g., the way to manage and allocate a virtual resource or a physical resource needs to be designed. Accordingly, design related to virtualized infrastructure (VI) is becoming part of NGN network evolution. Figure 6-1 illustrates the relationship between NGN architecture and virtualization-related design.

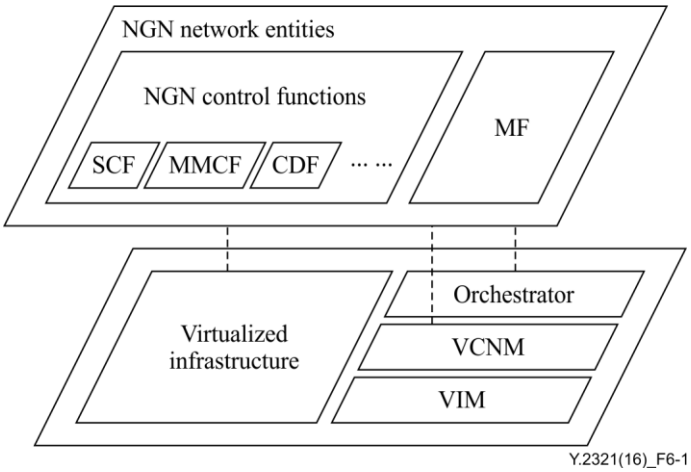

Figure 6-1 – The relationship between next generation network architecture and virtualization-related design

This Recommendation focuses on the design of the architecture related to virtualization of control network entities (VCN), including the VI, the management systems and the network entities. The reason for focusing on control network entities is that the performance degradation caused by virtualization has been proved acceptable by vendors and operators. Control functions in NGNs, such as SCFs, CDFs and mobility management and control functions (MMCFs), can be supported in this architecture. Since the architecture design is not dedicated to specified network entities, this Recommendation could also apply to other kinds of network entities, such as gateways, if they are virtualized.

7 Functional architecture for supporting virtualization of control network entities in next generation network evolution

This clause describes the functional architecture of VCN, FEs, and the reference points between the FEs.

7.1 Functional architecture

The functional architecture of the VCN identifies the functions shown in Figure 7-1.

- Virtualized infrastructure (VI)
- Virtualized infrastructure manager (VIM)
- VCN manager (VCNM)
- Virtualized Control Network Entity (VCNE)
- Orchestrator
- Management functions (MFs).

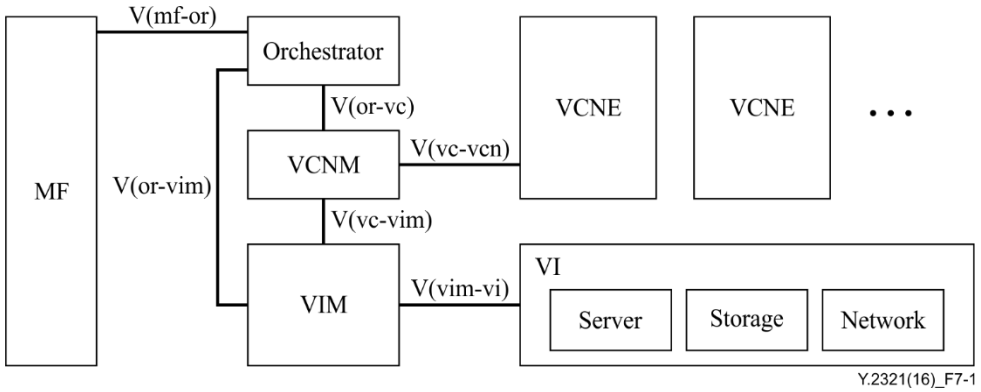

Figure 7-1 – Functional architecture for supporting VCN

These functions are used to support the virtualization of NGN control functions, i.e., NGN control functions are virtualized and deployed on to general infrastructure including computing, storage and network resource, so that these functions can be flexibly created, updated and deleted [ITU-T Y.2320]. NGN control functions (e.g., SCFs, transport control functions) are required to support the interface with the VCNM. The NGN MF needs to be enhanced to assist the virtualization of NGN control functions.

VI provides the execution environment to the VCNE. VCNE software can run on a hardware independent resource from VI with specific performance and portability requirements. It does not assume any specific control protocol between VI and VCNE.

Clause 7.1 describes the virtualized infrastructure, while clause 7.2 describes reference points V(or-vc), V(vc-vim), V(vc-vcn), V(vim-vi), V(or-vim) and V(mf-or).

7.1.1 Virtualized infrastructure

VI includes computing, network and storage resources that are operated and configured by the VIM. The resources in VI report the status information to VIM.

VI performs the following functions:

- Deploys the virtual machines (VMs) to physical server(s) according to the policy defined in the VCNM
- Resource usage monitoring for each of the VCNEs, every VM and all physical servers.
- Supports live migration
- Creates or deletes VMs
- Provides shared storage for the VM images
- Regenerates VMs automatically when active ones crash
- Adjusts the capacity of VMs (including computing, storage, and networking capacity).

7.1.2 Virtualized infrastructure manager

The VIM is responsible for controlling and managing the virtual infrastructure and includes:

- Infrastructure resource management

  - Booting or shutting down physical servers
  - Maintaining the resource availability data, i.e., maintaining the information of available physical servers
  - Resource reservation management
  - Resourcing recycling for VCNEs. If an allocated resource is released, for example, the system deletes existing VCNEs or removes VMs from VCNEs or reduces the computing, storage and networking capacity of VMs of VCNEs, etc., after which these resources can be available to meet demand from other VCNEs.
- VM lifecycle management
  - Configuring the VM, including the central processing unit (CPU), memory, network interface card (NIC) and storage
  - Creating, deleting, booting, hibernating and migrating VMs.
- Virtualized resource fault management
  - Monitoring failure or fault events (failure of shared storage, failure of fan or power module, etc.).
- Virtualized resource configuration management
  - Configuring networking parameters of servers or storage, such as IP address or virtual local area network (VLAN).
- Virtualized resource performance management
  - Monitoring the virtual resource status and resource utilization status (e.g., the CPU of the VM or memory utilization rate of the VM or the throughput of the network).

7.1.3 VCNM

The VCNM is responsible for VCNE lifecycle management, fault management, configuration management and performance management. The VCNM performs the following functions:

- VCNE-related virtualized resource management
  - Collects virtual resource status (e.g., resource utilization, failure or fault event) and service status (e.g., number of serving subscribers) pertaining to certain VCNE instances from the VIM
  - Supports prioritization of resource adjustment requests according to operation policy in order to manage potential resource adjustment conflicts
  - Provides VCNE templates that satisfy different capacity requirements of different control network entities
  - Judges whether the condition for applying for resources is reached, and sends resource requests to the orchestrator to apply for the resource from a specific resource pool or requests resources not available in specific resource pools if it requires the resource.
- VCNE lifecycle management
  - Performs on-boarding of VCNE packages.
  - VCNE instantiation, i.e., creating new VCNEs:
    - Generates the virtual resource requirement (including the required computing, network, storage) based on the virtualized control network entity descriptor (VCNED), and sends the requirements to the orchestrator.
  - VCNE instance scaling
    - Triggers the scaling according to the virtual resource information or service information it receives by adding or deleting VMs of the VCNE instance.

  - VCNE instance termination:
    - Terminates the VCNE instance and instructs the VIM to recycle the released virtual resource.
- VCNE fault management
  - Monitors the failure or fault event of the VCNE

  NOTE – Self-healing capability ensures service continuity when some of the servers or the VM stop. If the VCNE software does not have self-healing ability, the VCNM should inform the VCNE instance to migrate the services from the original VMs to other VMs. The self-healing mechanism can make the VCNE recover through the recovery or replacement of the VI. It may occur automatically under the control of the VCNM, and it may also be initiated directly by the VCNE itself.
- VCNE configuration management.
  - Configures deployment specific parameters of the VCNE such as the IP pool for subscribers.
- VCNE performance management.
  - Monitors virtualized resource status and service status of the VCNE (e.g., the traffic load, online accessed subscribers).

7.1.4 Virtualized control network entity

Besides the functions defined in the NGN architecture (SCFs, CDFs, etc.), the VCN should be enhanced to execute the following functions:

- Report the service volume (e.g., busy hours call attempts) to the VCNM.
  - Perform control network entity load monitoring and reporting to the VCNM.
- Satisfy high availability, including:
  - Failure of some of the related VMs does not affect the main function of the control network entity.
  - Adding new VMs does not affect the main function of the control network entity.
  - Adjusting VM capacity (including computing, storage and networking capacity) does not affect the main function of the control network entity.

7.1.5 Orchestrator

The orchestrator is responsible for global resource orchestration and lifecycle management of network service (NS) and includes:

- Global resource orchestration

  Procurement of virtual resource information from VIMs and maintenance of a global view of virtual resource.
- VCNE-related resource management

  Handling authorization requirements request from the VCNM(s) and selection of the VIM(s) that can satisfy these requirements with subsequent dispatch of resoursce requirements to the selected VIM(s). The orchestrator validates and authorizes resource requests from the VCNM, which may be rejected due to insufficient resource and other reasons.
- NS lifecycle management

  Rather than the VCNM, which supports lifecycle management of NS, the orchestrator manages the lifecycle of NSs that are basically composed of multiple NSs and connections among them.

- Interworking with NGN MFs

  The orchestrator supports interworking with MFs in the NGN and offers its own interfaces to be used by them. The orchestrator receives service requests from the MFs and instantiates NS according to these requests.

  NOTE – An NS is composed of a set of VCNEs.

7.1.6 Management functions

Besides the functions (e.g., fault management, configuration management, accounting management, security management and performance management) defined in the NGN architecture, the MFs should be enhanced to initiate the requests for:

- on-boarding of the VCNE package
- creation or termination of the VCNE or NS.

7.2 Reference points

7.2.1 V(or-vc)

The reference point between the orchestrator and the VCNM is mainly used as the VCN lifecycle management and to coordinate resource requests (e.g., for reservation, allocation, adjustment, release) involved in VCNE lifecycle management.

- Orchestrator→VCNM
  - Notifies the consequence of the resource request in VCNE lifecycle management, such as reservation, authorization, allocation, adjustment and release of the resource
  - Reports VCNE-related resource status.
- VCNM→Orchestrator
  - Resources requests in VCNE lifecycle management, such as reservation, authorization, allocation, adjustment and release of the resource. The orchestrator handles the resource request to make sure that the VCNM/VCN can obtain the requested resource. The request also contains the requirement description of resource deployment or specific requirements for the resource pool.

7.2.2 V(vc-vim)

- VCNM→VIM
  - The VCNM initiates the request for creation, query, monitor, release and migration of VM to the VIM
  - The VCNM sets the VM network topology and routing mechanism.
- VIM→VCNM
  - Reports the status of physical and virtual resource, performance data and configuration information.

7.2.3 V(vc-vcn)

- VCNM→VCNE
  - Notifies the change of VCNE lifecycle management, such as the consequence of VCN instantiation, scale out and scale in
  - Relays the performance, fault and status of VI to VCN.
- VCNE→VCNM
  - Initiates the VCNE lifecycle management request to the VCNM, such as scale out and scale in

– Reports the status of virtual resource, performance and fault messages to the VCNM.

7.2.4 V(vim-vi)

- VIM→VI
  - Executes the management of virtual resource through this reference point, such as creation, release, configuration and monitoring.
  - Performs live migration of the VM
  - Configures the network resource such as the routing table and quality of service (QoS) parameters.
- VI→VIM
  - Reports the configuration of related VI resource
  - Physical server reports the resource performance to the VIM, such as the CPU, memory and resource utilization of hard disk
  - Physical server reports the fault of the physical server to the VIM
  - Switch or Route reports the network conditions to the VIM.

7.2.5 V(or-vim)

- Orchestrator→VIM
  - Manages VCNE packages, such as on board and delete
  - Queries resources, including physical resources and virtualized resources
  - Monitors resource, including status information of physical resources and virtualized resources
  - Reserves resource
  - Resources performance management, including querying the performance (e.g., supported traffic throughput, CPU utilization rate) of physical resources and virtualized resources
  - Manages virtualized network resource, such as creating, updating and deleting the virtualized network.
- VIM→Orchestrator
  - Notifies resource changes, including changes of physical resources and virtualized resources
  - Manages alarm information, including reporting the alarm information fromof physical resources and virtualized resources.

7.2.6 V(mf-or)

- Orchestrator →MF
  - Manages NS descriptors and associated descriptors, such as VCNE and virtual link descriptor (VLD). Typical operations are for on-boarding, querying, disabling, enabling, updating and deleting these descriptors
  - Provides performance management (measurement results collection and notifications) related to NSs
  - Allows the orchestrator to provide alarms related to the NSs visible to the consumer
  - Allows the management of VCNE packages including operations for on-boarding, querying, disabling, enabling and deleting VCNE packages.
- MF → Orchestrator

– Allows the MF to subscribe to NS lifecycle change notifications, and the orchestrator provides such notifications to the subscriber.

8 Security considerations

This Recommendation requires no specific considerations and aligns with the security requirements specified in [ITU-T Y.2701].

Annex A

Information flows for virtualization of a control network

(This annex forms an integral part of this Recommendation.)

The messages and flows used in this annex are representative and are not meant to indicate any particular protocol.

A.1 On-boarding VCNE package

See Figure A.1.

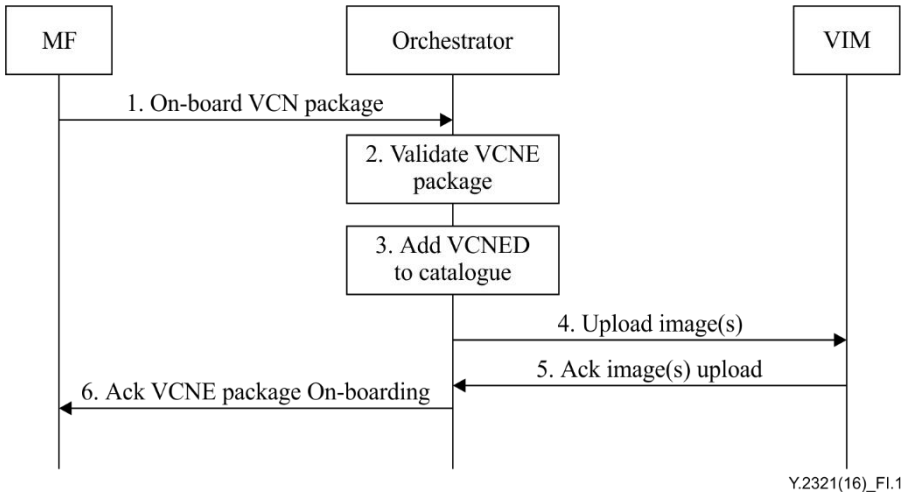

Figure A.1 – VCNE package on-boarding information flow

The main steps for VCNE package on-boarding are:

1 The MF sends on-boarding request to the orchestrator and the VCNE package is submitted to the orchestrator.

2 The orchestrator validates the accuracy and integrity of the VCNE package, e.g., checking for the existence of mandatory elements in the VCNED.

3 The orchestrator adds the VCNED to the catalogue.

4 The orchestrator uploads the image(s) to the VIM.

5 The VIM acknowledges the successful uploading of the image.

6 The orchestrator acknowledges the VCN package on-boarding to the MF.

A.2 Delete VCNE package

See Figure A.2.

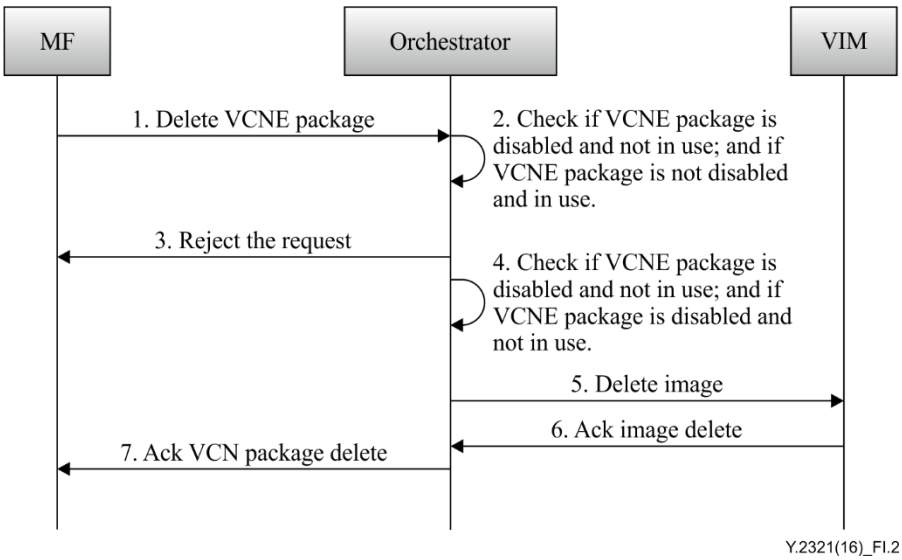

Figure A.2 – Delete VCNE package information flow

The main steps for VCNE package deletion are:

1 The MF request to delete the VCNE package is sent to the orchestrator.

2 The orchestrator checks whether the VCNE package is disabled and not in use. If the VCNE package is not disabled and still in use, the orchestrator sets the VCNE package to deletion pending.

3 The request is rejected.

4 If VCNE package is disabled and not in use, then the orchestrator asks the catalogue to remove all the versions of the VCNE package. The catalogue then removes it/them.

5 The orchestrator deletes image(s) from the VIM(s) stored.

6 The VIM acknowledges the successful deletion of the image(s).

7 The orchestrator acknowledges the VCNE package deletion request.

A.3 VCNE instantiation

See Figure A.3.

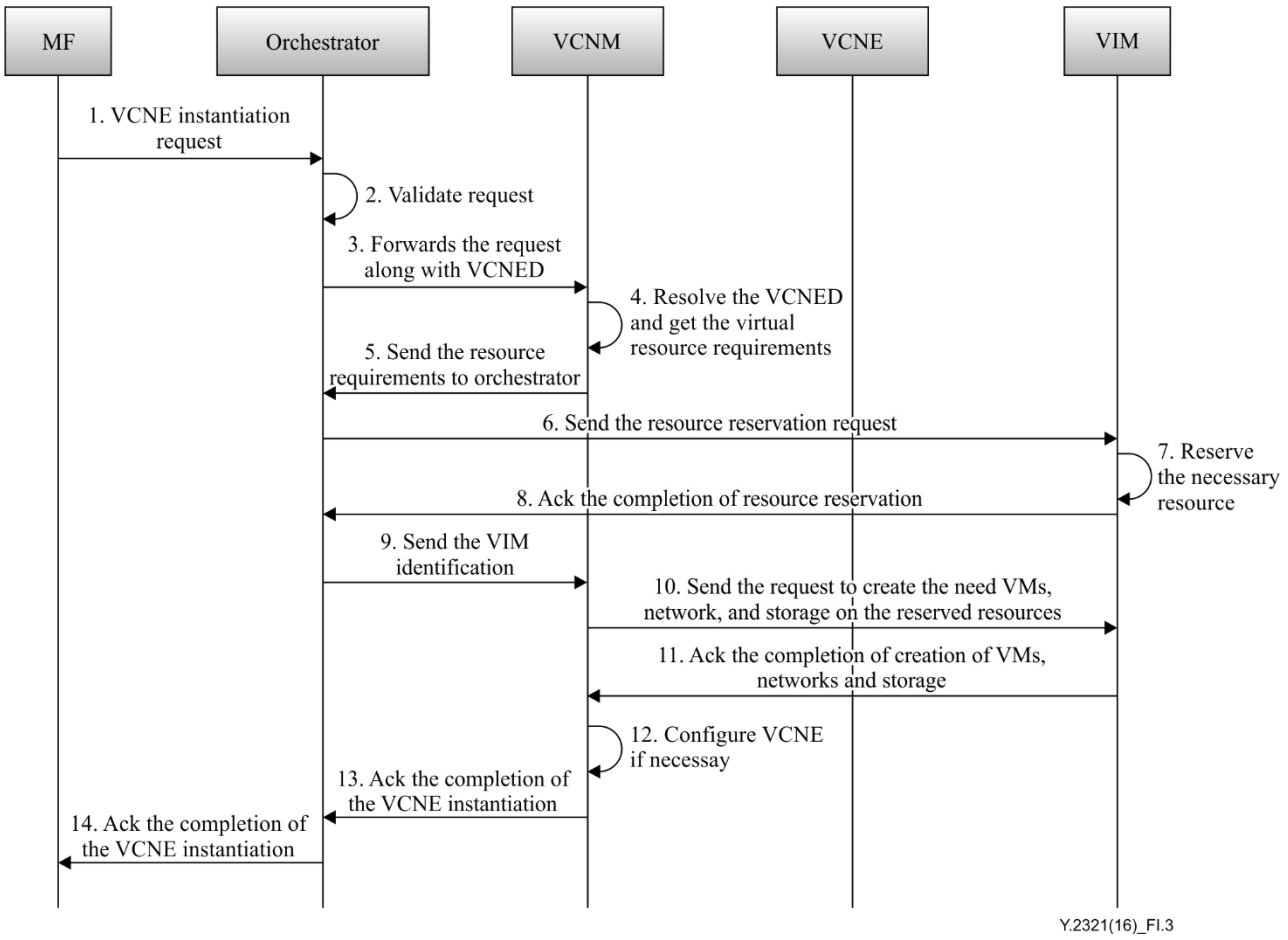

Figure A.3 – VCNE instantiation information flow

The main steps for VCNE instantiation are:

1 The MF sends a request to instantiate a new VCNE and the request indicates the selected VCNED.

2 The orchestrator validates the request.

3 The orchestrator forwards the request to the VCNM along with the VCNED or its identification.

4 The VCNM resolves the VCNED and gets the virtual resource requirements.

5 The VCNM sends the resource authorization request to the orchestrator. The request can include three types of information. The first type of information indicates no specific VIM preference. The second type of information can include the identification of preferred VIMs (i.e., the VCNM wants to request the resource of these VIMs). The third type of information includes the identification of not preferred VIMs (i.e., the VCNM does not want to request the resource of these VIMs).

6 The orchestrator selects a VIM that can satisfy the resource authorization requirements, and sends the resource reservation request to the VIM.

7 The VIM reserves the necessary resource, including computing, networking and storage.

8 The VIM acknowledges the completion of resource reservation to the orchestrator.

9 The orchestrator sends the resource authorization response with the VIM identification to the VCNM, which identify the VIM. The VCNM can then request virtualized resource.

10 The VCNM sends a request to the VIM to create the needed VMs, network and storage on the reserved resources.

11 The VIM acknowledges the completion of creation of VMs, network and storage.

12 The VCNM configures the VCNE with any VCNE specific parameters, if necessary.

13 The VCNM acknowledges the completion of the VCNE instantiation to the orchestrator.

14 The orchestrator acknowledges the completion of the VCNE instantiation to the MF.

A.4 Scale out of VCNE

See Figure A.4.

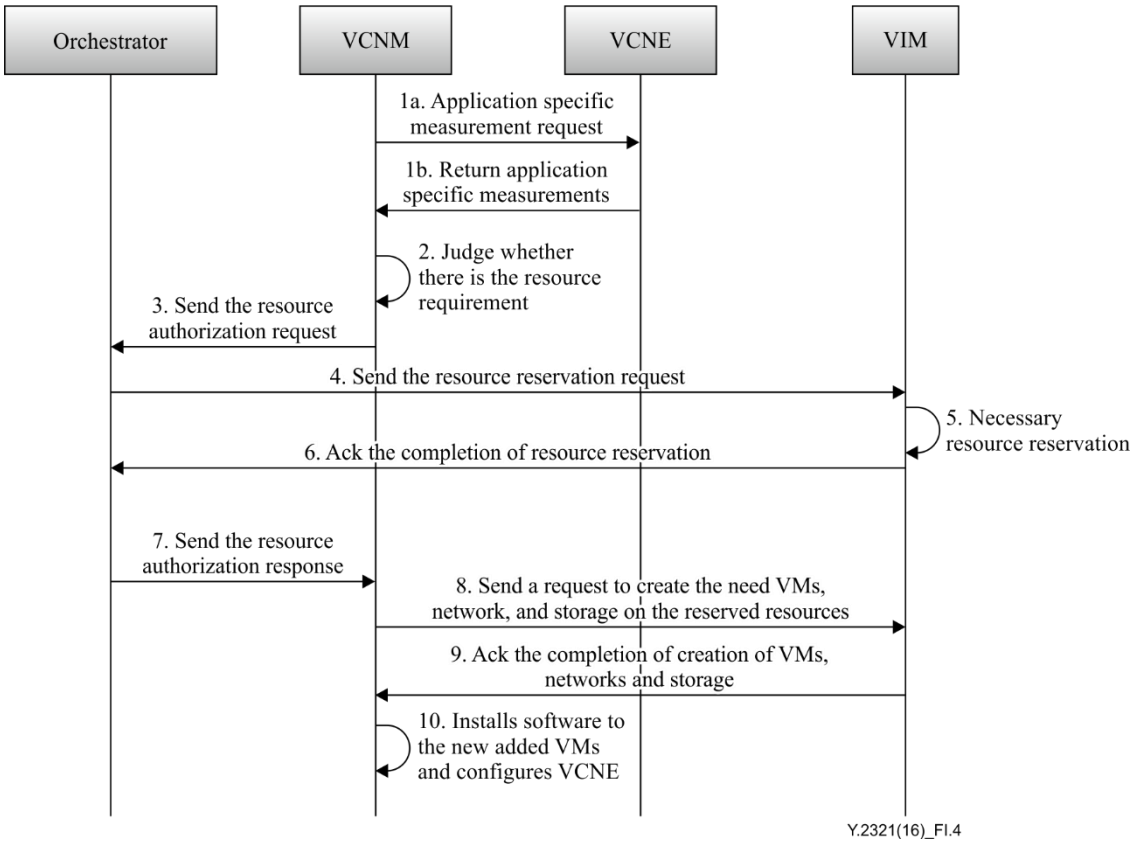

Figure A.4 – VCNE scale out information flow

The main steps for VCNE scale out are:

1 The VCNM collects application specific measurements from the VCNE.

2 The VCNM judges whether there is a resource requirement based on the policy and the result of the measurement information, e.g., whether there is capacity shortage to require expansion (add more resources).

3 If the VCNM decides the VCNE has the resource requirement, the VCNM sends a resource authorization request to the orchestrator. The information can be of three types. The first type of information indicates no specific VIM preference. The second type of information can include the identification of preferred VIMs (i.e., the VCNM wants to request the resource of these VIMs). The third type of information includes the identification of not preferred VIMs (i.e., the VCNM does not want to request the resource of these VIMs).

4 The orchestrator selects a VIM that can satisfy the resource authorization requirements and sends the resource reservation request to the VIM.

5 The VIM reserves the necessary resource, including computing, networking and storage.

6 The VIM acknowledges the completion of resource reservation to the orchestrator.

7 The orchestrator sends the resource authorization response with the VIM identification to the VCNM, which identify the VIM, the VCNM can then request virtualized resource.

8 The VCNM sends a request to the VIM to create the needed VMs, network and storage on the reserved resources.

9 The VIM acknowledges the completion of creation of VMs, network and storage.

10 The VCNM configures the VCNE with any VCNE specific parameters, if necessary.

A.5 Scale in of VCNE

See Figure A.5.

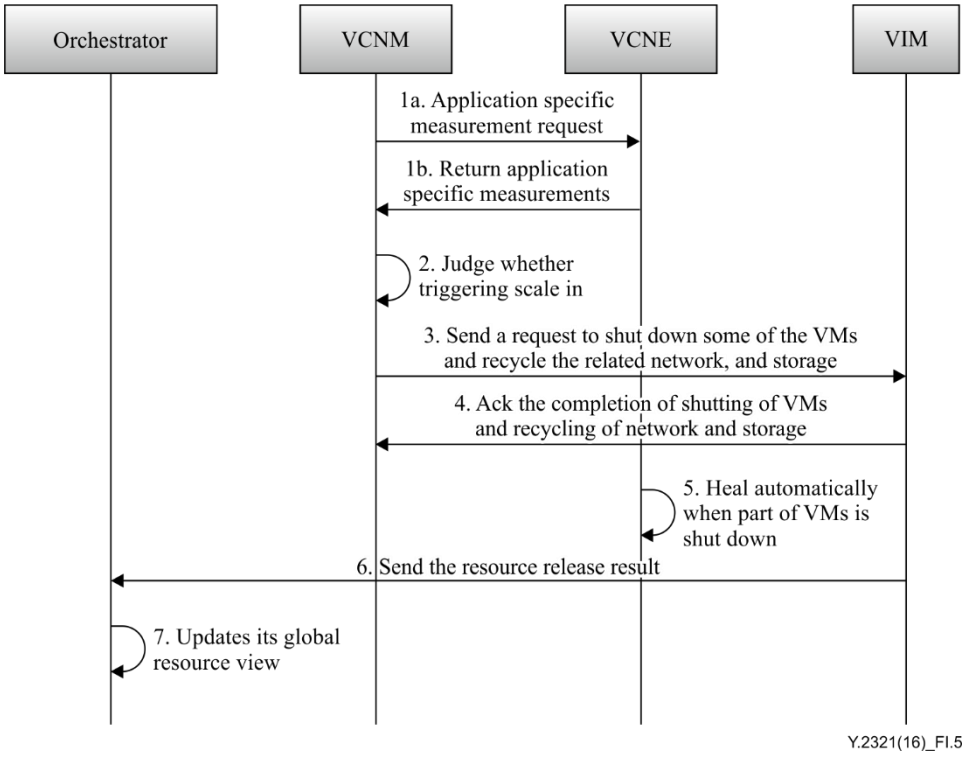

Figure A.5 – VCNE scale in information flow

The main steps for VCNE scale in are:

1 The VCNM collects application specific measurements from the VCNE.

2 The VCNM detects the capacity to judge whether triggering scale in (release of resources).

3 If triggering scale in, the VCNM sends a request to the VIM to shut down some of the VMs and recycle the related network, and storage.

4 The VIM acknowledges the completion of shutting down of VMs and recycling of network and storage.

5 The VCNE heals automatically when some of the VMs are shut down.

6 The VIM sends the resource release result to the orchestrator.

7 The orchestrator updates its global resource view.

A.6 Scaling of VCNE manually

See Figure A.6.

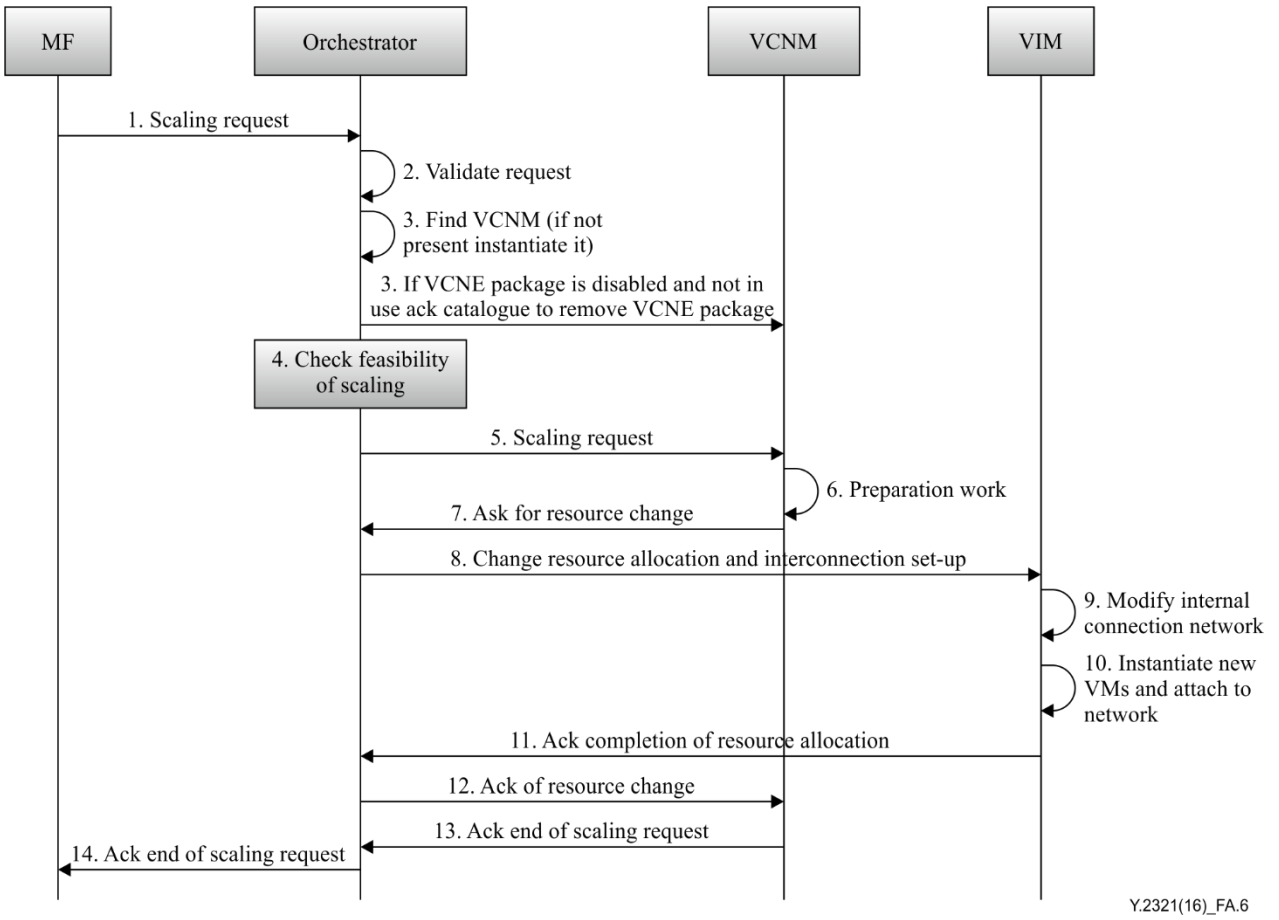

Figure A.6 – Scale out of VCNE manually information flow

The main steps for the VCNE scaling manually are:

1 The orchestrator receives the scaling request from the MF.

2 The orchestrator validates the request.

3 The orchestrator finds the VCNM relevant to this VCNE type.

4 Optionally, the orchestrator runs a feasibility check of the VCNE scaling request to reserve resources before doing the actual scaling.

5 The orchestrator sends the scaling request to the VCNM with the scaling data and, if step 4 has been done and it is feasible.

6 The VCNM executes any needed preparation work: requests validation, parameter validation. If step 4 was done then the VCNM skips step 6.

7 The VCNM calls the orchestrator for resource change.

8 The orchestrator requests allocation of changed resources (compute, storage and network) needed for the scaling request from the VIM.

9 The VIM modifies the internal connectivity of network as needed.

10 The VIM creates and starts the needed new compute (VMs) and storage resources and attaches new instantiated VMs to the internal connectivity network.

11 Acknowledgement of completion of resource change to the orchestrator.

12 The orchestrator acknowledges the completion of the resource change to the VCNM.

13 The VCNM configures the scaled VCNE as necessary. The VCNM acknowledges the end of the scaling request to the orchestrator.

14 The orchestrator acknowledges the end of the scaling request to the MF.

A.7 Termination of VCNE

See Figure A.7.

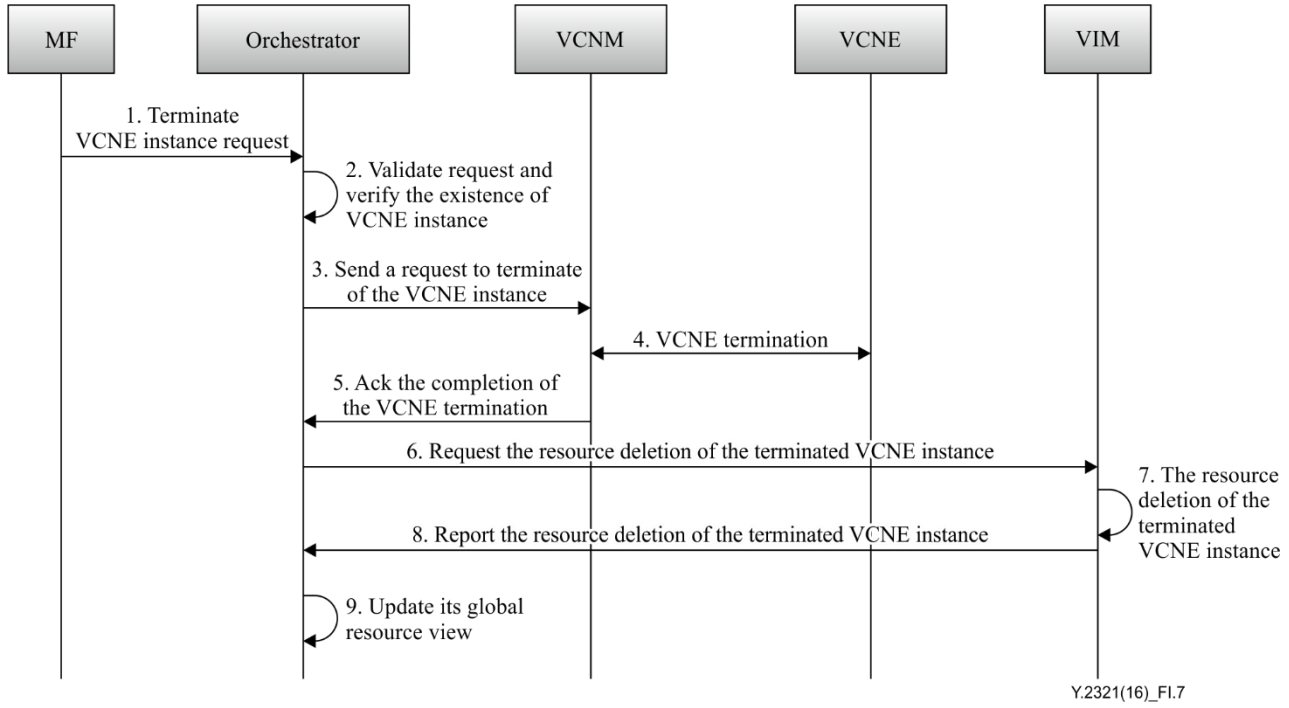

Figure A.7 – VCNE termination information flow

The main steps for the VCNE termination are:

1 The MF requests the orchestrator to terminate an existing VCNE instance.

2 The orchestrator validates the request (including the authorization of the MF) and verifies that the VCNE instance exists.

3 The orchestrator sends a request to the VCNM to terminate the VCNE instance.

4 The VCNM terminates the VCNE. This step may include a graceful shutdown of the VCNE possibly in coordination with another VCNE instance.

5 The VCNE is terminated, and the VCNM acknowledges the completion of the VCNE termination to the orchestrator.

6 The orchestrator sends the request to delete the resource of the terminated VCNE instance.

7 The VIM deletes the resource occupied by the terminated VCNE instance.

8 The VIM reports the resource deletion of the terminated VCNE instance to the orchestrator.

9 The orchestrator updates its global resource view.

A.8 Network service instantiation flows

See Figure A.8.

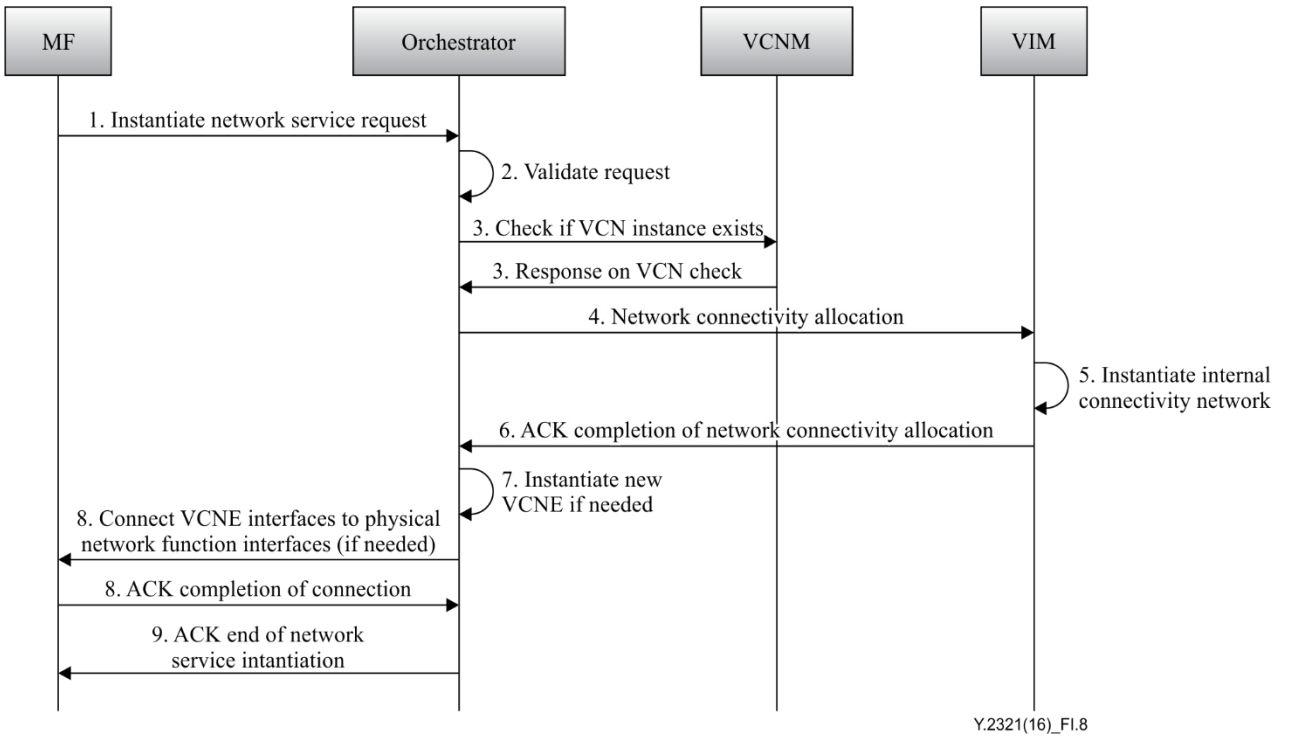

Figure A.8 – Network service instantiation information flow

The main steps for NS instantiation are:

1 The orchestrator receives a request to instantiate a new NS.

2 The orchestrator validates the request, both validity of request (including validating that the MF is authorized to issue this request) and validation of the parameters passed for technical correctness and policy conformance. If the NS contains multiple VCNEs and the forwarding graphs, policy rules might result in only a subset being valid for a given NS instance.

3 For each VCNE instance needed in the NS, the orchestrator checks with the VCNM if an instance matching the requirements exists already. If such a VCNE instance exists, it will be used as part of the NS.

4 The orchestrator requests instantiation of the network connectivity from the VIM. Some of the network connectivity between the VCNEs might already exist and might only need to be extended.

5 The VIM instantiates the connectivity network needed for the NS.

6 The VIM acknowledges completion.

7 Assuming the list of VCNE instances to be provisioned is not empty, the orchestrator instantiates the new VCNE instances needed.

8 If needed, the orchestrator requests the MF to connect the VCNE external interfaces to physical network function (PNF) interfaces, in which case the MF acknowledges completion of the connection.

9 The orchestrator acknowledges the completion of the NS instantiation.

A.9 Network service instance scale-out

See Figure A.9.

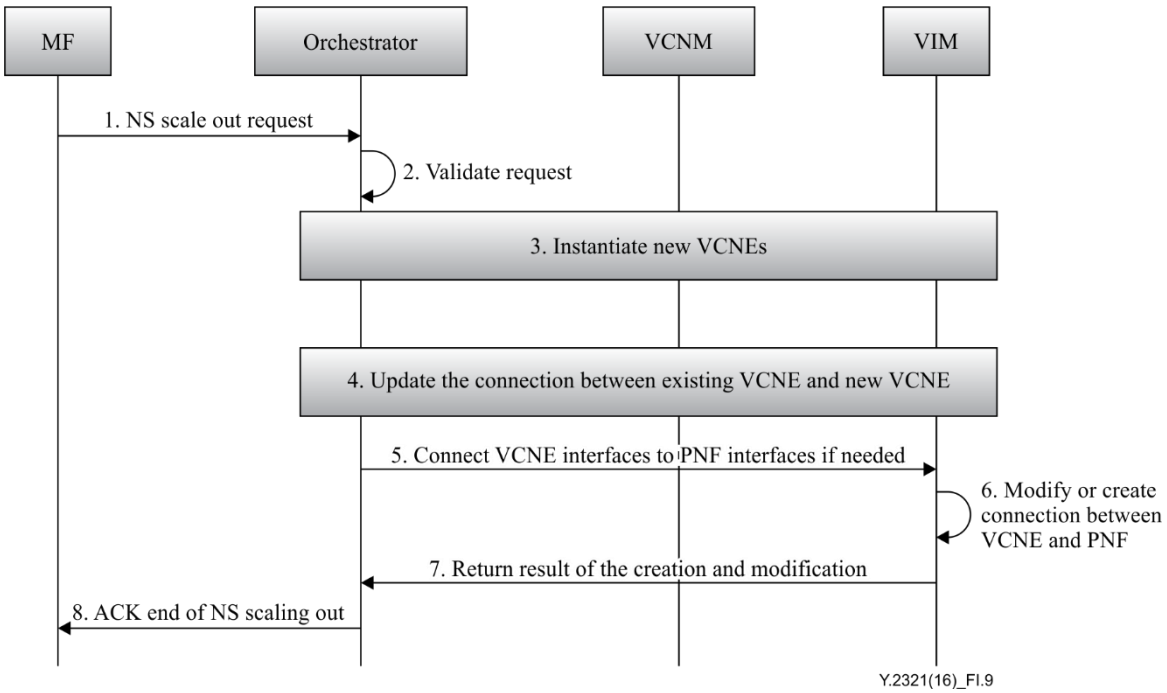

Figure A.9 – Network service scale out information flow

The main steps for NS instance scale out are:

1 The MF requests the NS to be scaled out to a new deployment flavour that is already present in the preloaded network service descriptor (NSD).

2 The orchestrator validates the request, both validity of request (including validating that the MF is authorized to issue this request) and validation of the parameters passed for technical correctness and policy conformance. The orchestrator correlates the request with the on boarded NSD in the NS catalogue.

3 VCNE instantiation flow is triggered.

4 Connections between the existing VCNE and new VCNE can be updated.

5 The orchestrator requests the VIM to allocate the changed resources.

6 The VIM allocates the interconnectivity accordingly.

7 The VIM returns the result of the operation to the orchestrator.

8 The orchestrator acknowledges the end of the scaling request to the requester.

A.10 Network service instance scale-in

See Figure A.10.

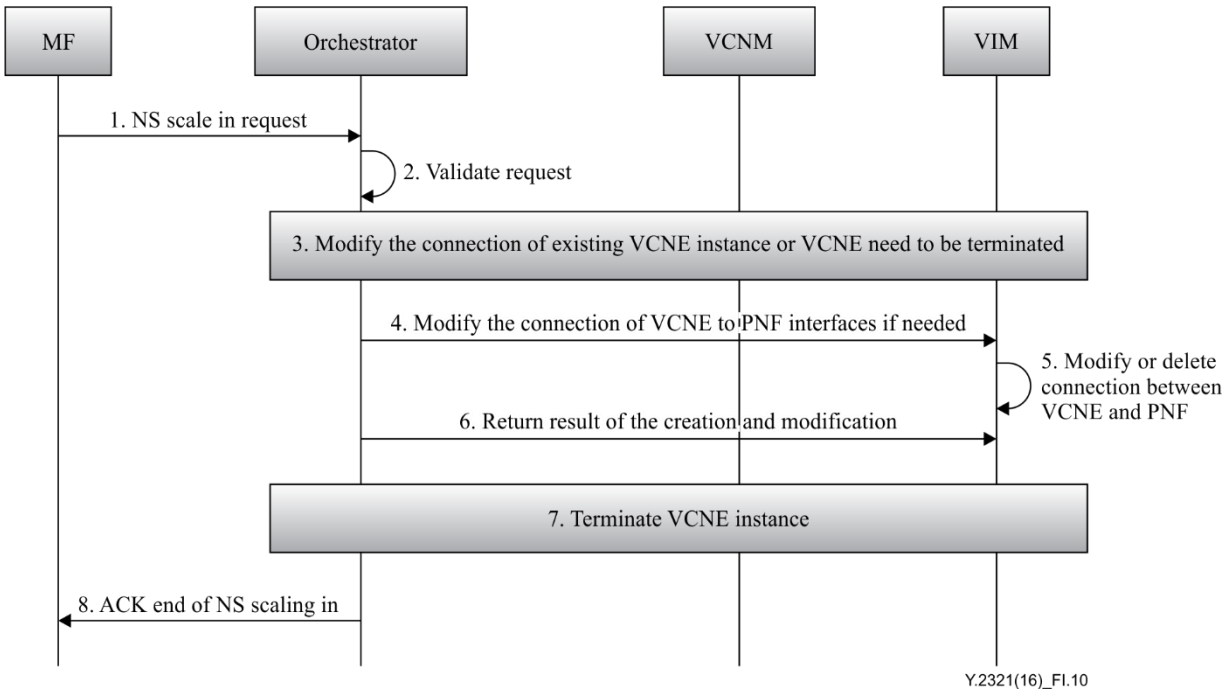

Figure A.10 – Network service instance scale in information flow

The main steps for NS scale in are:

1 The MF requests the NS to be scaled in to a new deployment flavour that is already present in the preloaded NSD.

2 The orchestrator validates the request, both validity of request (including validating that the MF is authorized to issue this request) and validation of the parameters passed for technical correctness and policy conformance.

3 Modify the connection of the existing VCNE instance or VCNE need to be terminated.

4 The orchestrator requests the VCNE to modify or delete the changed connection between the VCNE and PNF.

5 The VIM modifies or deletes the interconnectivity accordingly.

6 The VIM returns the result of the operation to the orchestrator.

7 VCNE instance termination flow.

8 The orchestrator acknowledges the end of the scaling request to the requester.

A.11 Network service instance termination flow

See Figure A.11.

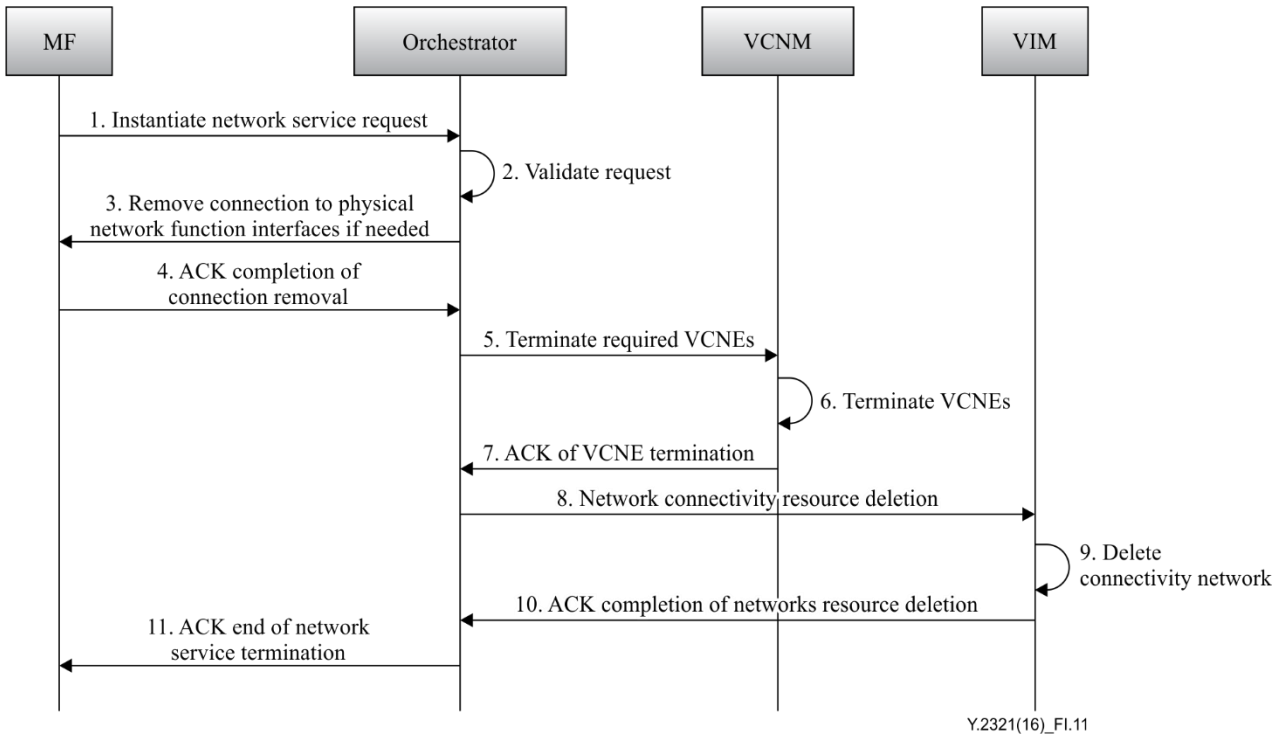

Figure A.11 – Network service instance termination information flow

The main steps for termination of an NS instance are:

1 The orchestrator receives the request to terminate an NS instance.

2 The orchestrator validates the request. It verifies the validity of the request (including the MF's authorization) and verifies that the network service instance exists.

3 If needed, orchestrator requests MF to remove the connections to physical network function interfaces.

4 If connections were removed, the MF acknowledges completion of the removal of connections.

5 The orchestrator requests the VCNM to terminate any VCNE instances that were instantiated along with the NS instantiation provided, which are not used by another NS.

6 The VCNM terminates the required VCNEs. This step might include some graceful termination of the VCNEs involved, possibly in coordination with other management entities or the VCNEs themselves.

7 The VCNM acknowledges completion of the termination request to the orchestrator.

8 Using information kept for this NS instance, the orchestrator requests deletion of network connectivity for this NS instance.

NOTE – Some network connectivity might have been present before the instantiation of the NS. This connectivity is not deleted.

9 The VIM deletes the connectivity network for this NS instance.

10 The VIM acknowledges the completion of resource deletion to the orchestrator.

11 The orchestrator acknowledges the completion of the NS instance termination. If the NSD is in deletion pending and there is no NS instance using it, the orchestrator asks the catalogue to remove the corresponding version(s) of the NSD. The catalogue then removes it.

SERIES OF ITU-T RECOMMENDATIONS

Series A	Organization of the work of ITU-T
Series D	Tariff and accounting principles and international telecommunication/ICT economic and policy issues
Series E	Overall network operation, telephone service, service operation and human factors
Series F	Non-telephone telecommunication services
Series G	Transmission systems and media, digital systems and networks
Series H	Audiovisual and multimedia systems
Series I	Integrated services digital network
Series J	Cable networks and transmission of television, sound programme and other multimedia signals
Series K	Protection against interference
Series L	Environment and ICTs, climate change, e-waste, energy efficiency; construction, installation and protection of cables and other elements of outside plant
Series M	Telecommunication management, including TMN and network maintenance
Series N	Maintenance: international sound programme and television transmission circuits
Series O	Specifications of measuring equipment
Series P	Telephone transmission quality, telephone installations, local line networks
Series Q	Switching and signalling, and associated measurements and tests
Series R	Telegraph transmission
Series S	Telegraph services terminal equipment
Series T	Terminals for telematic services
Series U	Telegraph switching
Series V	Data communication over the telephone network
Series X	Data networks, open system communications and security
Series Y	**Global information infrastructure, Internet protocol aspects, next-generation networks, Internet of Things and smart cities**
Series Z	Languages and general software aspects for telecommunication systems

Printed in Switzerland
Geneva, 2017