教育部「5G行動寬頻人才培育跨校教學聯盟計畫」
5G行動網路協定與核網技術聯盟中心

行動邊緣計算
可推廣教材模組

單元-06：邊緣計算與網路功能虛擬化：功能與架構

授課教師：萬欽德

國立高雄科技大學 電腦與通訊工程系

Definitions and Abbreviations

- **ME app VNF:** mobile edge application that appears like a VNF towards the ETSI NFV MANO components

- Abbreviations
  - DNS Domain Name System
  - DOPFR Dynamic Optimization of Packet Flow Routing
  - EM Element Manager
  - ETSI European Telecommunications Standards Institute
  - GS Group Specification
  - GTP GPRS Tunneling Protocol
  - HOT Heat Orchestration Template
  - IP Internet Protocol

Definitions and Abbreviations

- LCM Life Cycle Management
- MANO Management and Orchestration
- ME Mobile Edge
- MEAO Mobile Edge Application Orchestrator
- MEC Mobile Edge Computing
- MEO Mobile Edge Orchestrator
- MEPM Mobile Edge Platform Manager
- MEPM-V Mobile Edge Platform Manager - NFV
- NFP Network Forwarding Path
- NFV Network Functions Virtualization
- NFVI Network Functions Virtualization Infrastructure
- NFVO Network Functions Virtualization Orchestrator

Definitions and Abbreviations

- NS Network Service
- NSD Network Service Descriptor
- OASIS Organization for the Advancement of Structured Information Standards
- OSS Operations Support System
- PM Performance Management
- PNF Physical Network Function
- PNFD Physical Network Function Descriptor
- PoP Point of Presence
- QCI Quality Class Indicator
- SFC Service Function Chaining
- SPID Subscriber Profile ID

Definitions and Abbreviations

- TEID Tunnel Endpoint ID
- TOSCA Topology and Orchestration Specification for Cloud Applications
- UE User Equipment
- VIM Virtualized Infrastructure Manager
- VL Virtual Link
- VM Virtual Machine
- VNF Virtualized Network Function
- VNFC VNF Component
- VNFD VNF Descriptor
- VNFFG VNF Forwarding Graph
- VNFM Virtual Network Function Manager
- YAML YAML Ain't Markup Language

MEC deployments in NFV environments

- Mobile network operators are expected to virtualize their networks using Network Functions Virtualization (NFV), and want to use the virtualization infrastructure to consolidate network elements (Virtualized Network Functions - VNFs), Mobile Edge Computing (MEC) components and Mobile Edge (ME) applications on top of that infrastructure.
- Sharing the introduced elements (infrastructure, but also management functions) allows to make maximum use of the investments into virtualization.

Reference Architecture - Overview and Assumptions (1/2)

- The ME platform is deployed as a VNF. For that purpose, the procedures defined by ETSI NFV are used. It is not expected that these procedures need to be modified for use with ETSI MEC.
- The ME applications appear like VNFs towards the ETSI NFV MANO components. This allows re-use of ETSI NFV MANO functionality. It is, however, expected that ETSI MEC might not use the full set of MANO functionality, and requires certain additional functionality.
- Such a specific ME application is denoted by the name "**ME app VNF**".

Reference Architecture - Overview and Assumptions (2/2)

- The virtualization infrastructure is deployed as a NFVI and its virtualized resources are managed by the VIM.
- For that purpose, the procedures defined by ETSI NFV Infrastructure specifications, i.e. ETSI GS NFV-INF 003, ETSI GS NFV-INF 004, ETSI GS NFV-INF 005, can be used.
- It is not expected that these procedures need to be modified for use with ETSI MEC.

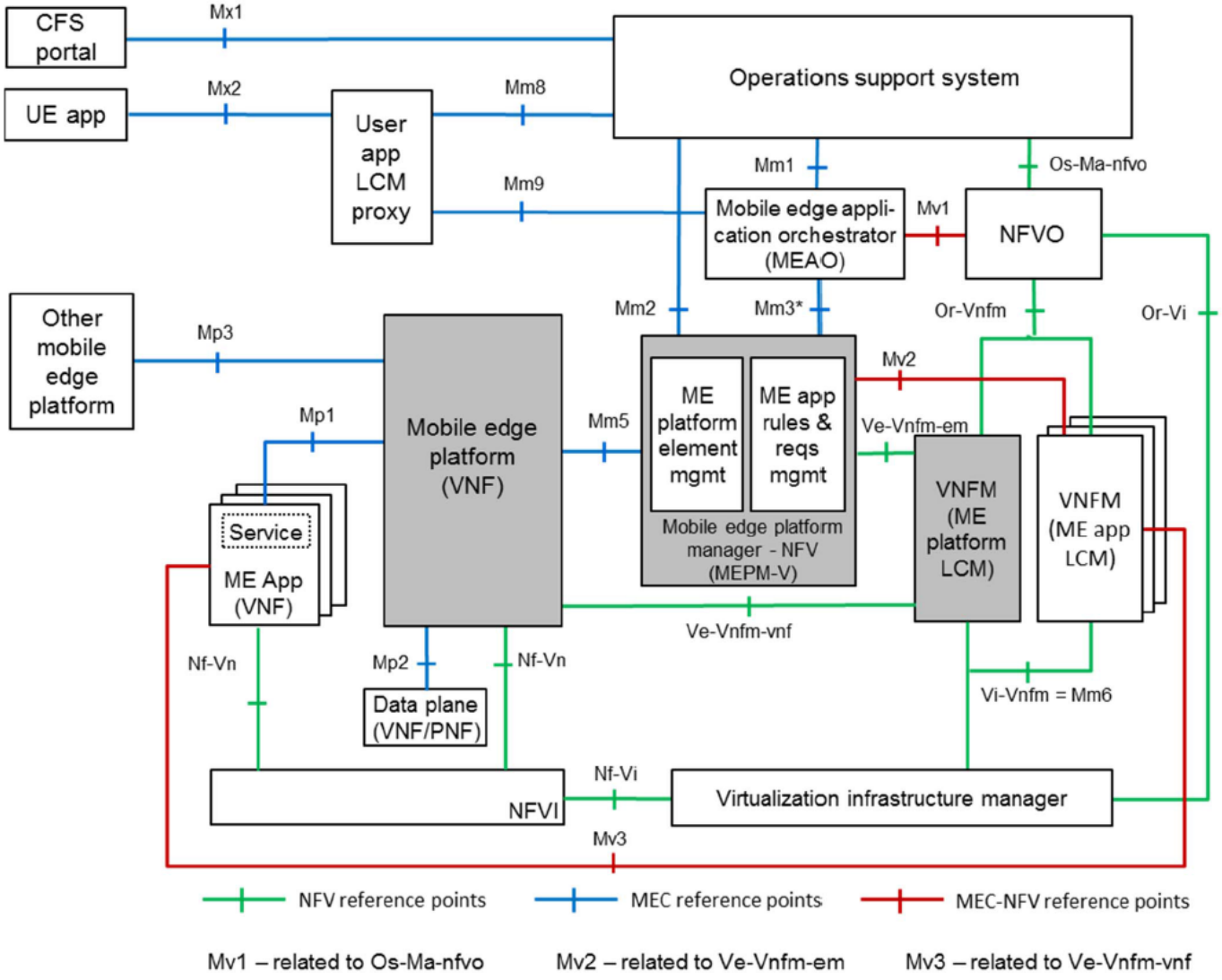
MEC reference architecture in NFV environments

- It is assumed that the **ME app VNFs** will be managed like individual VNFs, allowing that a MECinNFV deployment can delegate certain orchestration and Life Cycle Management (LCM) tasks to the NFVO and VNFM functional blocks, as defined by ETSI NFV MANO.
- One goal of the present study is to describe that actual mapping, to elaborate on a high level how MEC orchestration procedures and ME application LCM procedures can be realized in such an environment.

MEC reference architecture in NFV environments (cont'd)

- The Mobile Edge Platform Manager (MEPM), as defined in the MEC reference architecture ETSI GS MEC 003, is transformed into a "Mobile Edge Platform Manager - NFV" (MEPM-V) that delegates the LCM part to one or more VNFM(s).
- The Mobile Edge Orchestrator (MEO), as defined in the MEC reference architecture ETSI GS MEC 003, is transformed into a "Mobile Edge Application Orchestrator" (MEAO) that uses the NFVO for resource orchestration, and for orchestration of the set of ME app VNFs as one or more NFV Network Services (NSs).

ETSI MEC reference architecture in an ETSI NFV environment



ETSI MEC reference architecture in an ETSI NFV environment (cont'd)

NOTE 1: It is assumed that the Mobile Edge Platform VNF, the MEPM-V and VNFM (ME platform LCM) will be deployed as a single package as per the ensemble concept in 3GPP TR 32.842, or that the VNFM is a Generic VNFM as per ETSI GS NFV-IFA 009 and the Mobile Edge Platform VNF and the MEPM-V are provided by a single vendor.

NOTE 2: The Mp1 reference point between an ME application and the ME platform is optional for the ME application, unless it is an application that provides and/or consumes a ME service (ETSI GS MEC 003, Figure 6-1).

NOTE 3: The Mm3* reference point between MEAO and MEPM-V is based on the Mm3 reference point, as defined by ETSI GS MEC 003. Changes will be needed to this reference point to cater for the split between MEPM-V and VNFM (ME applications LCM).

New reference points (Mv1, Mv2 and Mv3)

Mv1: This reference point connects the MEAO and the NFVO. It is related to the Os-Ma-nfvo reference point, as defined in ETSI NFV.

Mv2: This reference point connects the VNF Manager that performs the LCM of the ME app VNFs with the MEPM-V to allow LCM related notifications to be exchanged between these entities. It is related to the Ve-Vnfm-em reference point as defined in ETSI NFV, but will possibly include additions, and might not use all functionality offered by Ve-Vnfm-em.

Mv3: This reference point connects the VNF Manager with the ME app VNF instance, to allow the exchange of messages e.g. related to ME application LCM or initial deployment-specific configuration. It is related to the Ve-Vnfm-vnf reference point, as defined in ETSI NFV, but will possibly include additions, and might not use all functionality offered by Ve-Vnfm-vnf.

Reference points defined by ETSI NFV

- **Nf-Vn**: This reference point connects each ME app VNF with the NFVI.
- **Nf-Vi**: This reference point connects the NFVI and the VIM.
- **Os-Ma-nfvo**: This reference point connects the OSS and the NFVO. It is primarily used to manage NSs, i.e. a number of VNFs connected and orchestrated to deliver a service.
- **Or-Vnfm**: This reference point connects the NFVO and the VNFM. It is primarily used for the NFVO to invoke VNF LCM operations.
- **Vi-Vnfm**: This reference point connects the VIM and the VNFM. It is primarily used by the VNFM to invoke resource management operations to manage the cloud resources that are needed by the VNF. It is assumed in a NFV-based MEC deployment that this reference point corresponds 1:1 to Mm6.
- **Or-Vi**: This reference point connects the NFVO and the VIM. It is primarily used by the NFVO to manage cloud resources capacity.

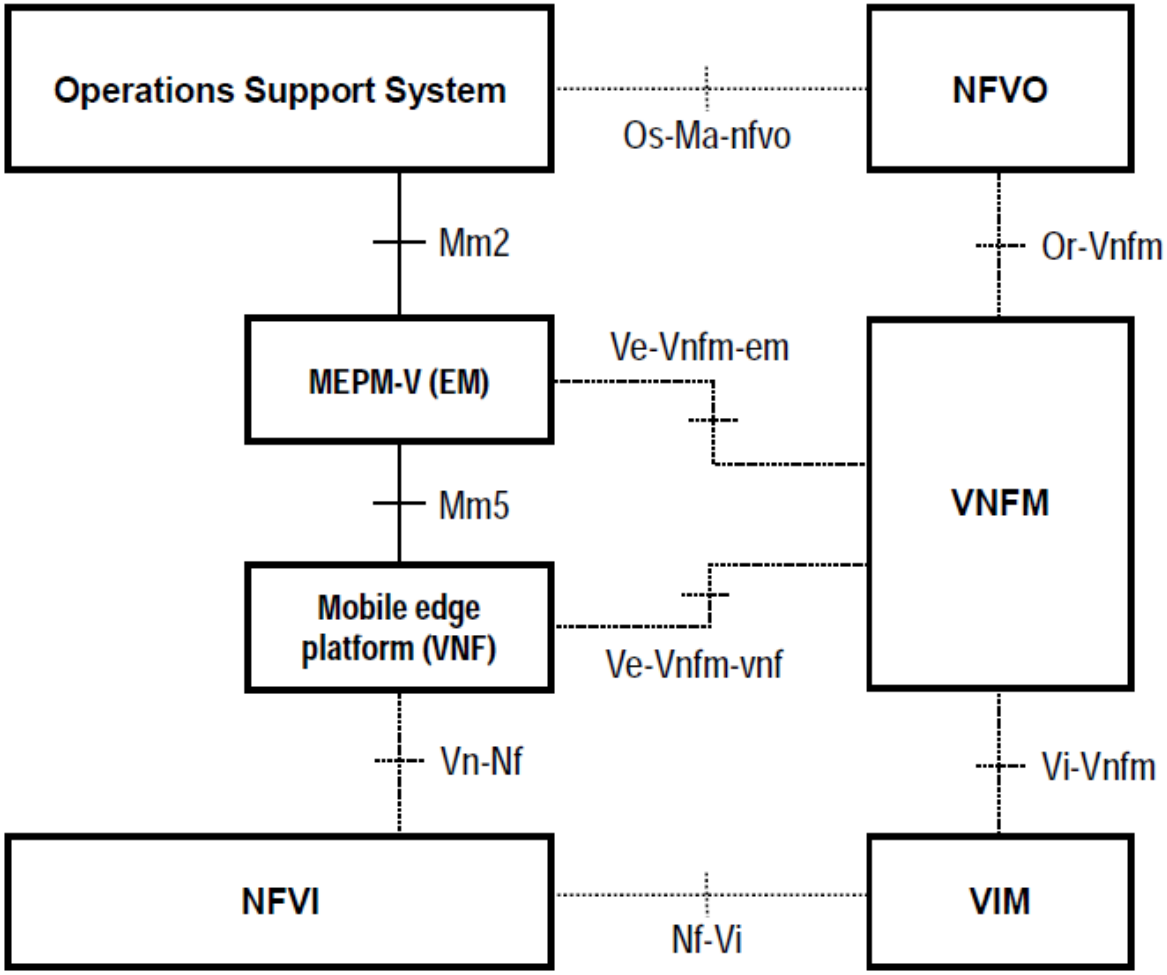
Realization of the mobile edge platform as a VNF

It is assumed that the ME platform will be realized as a VNF and will be managed according to ETSI NFV procedures. It is not assumed that ETSI MEC needs to define any modification to this.

This means:

- the MEPM-V will act as the Element Manager (EM) of the ME platform VNF;
- a VNF Manager, according to ETSI NFV (e.g. Specific VNFM, Generic VNFM), is used to perform LCM of the ME platform VNF;
- the scope of the Mp2 reference point will need to be redefined. ETSI GS MEC 003 states that this reference point is considered outside the scope of standardization but the introduction of the ME platform as a VNF introduces a potential multivendor deployment of the ME Platform VNF and the NFVI, which contains the Data Plane.

Management of the ME platform as a VNF



Management of the ME platform as a VNF: Reference Points

Ve-Vnfm-em: This reference point connects the VNF Manager (VNFM) that manages the lifecycle of the ME platform with the Mobile Edge Platform Manager - NFV (MEPM-V). It is the Ve-Vnfm-em reference point as defined in ETSI NFV. Since the Mobile Edge Platform VNF is considered as a network function, it is not expected that there are any impacts to the Ve-Vnfm-em procedures as defined by ETSI NFV.

Ve-Vnfm-vnf: This reference point connects the VNFM that manages the lifecycle of the ME platform with the Mobile Edge Platform VNF. It is the Ve-Vnfm-vnf reference point as defined in ETSI NFV. Since the Mobile Edge Platform VNF is considered as a network function, it is not expected that there are any impacts to the Ve-Vnfm-vnf procedures as defined by ETSI NFV.

Management of the ME platform as a VNF: Reference Points (cont'd)

- **Nf-Vn**: This reference point connects the Mobile Edge Platform VNF and the NFVI.
- **Nf-Vi**: This reference point connects the NFVI and the VIM.
- **Os-Ma-nfvo**: This reference point connects the OSS and the NFVO. It is primarily used to manage NSs, i.e. a number of VNFs connected and orchestrated to deliver a service.
- **Or-Vnfm**: This reference point connects the NFVO and the VNFM that manages the lifecycle of the ME platform. It is primarily used for the NFVO to invoke VNF LCM operations.
- **Vi-Vnfm**: This reference point connects the VIM and the VNFM that manages the lifecycle of the ME platform. It is primarily used by the VNFM to invoke resource management operations to manage the cloud resources that are needed by the VNF.
- **Or-Vi**: This reference point connects the NFVO and the VIM. It is primarily used by the NFVO to manage cloud resources capacity.

Realization of the Data Plane

- When MEC is deployed in a NFV environment, there are two different possibilities to realize the Data Plane, both are valid and need to be supported.

- **Option 1**: Realize the Data Plane as a PNF or VNF or combination thereof, and connect it to the NS that contains the ME app VNFs. In this option, Mp2 is kept as a MEC-internal reference point also in the NFV-based deployment of MEC; the option is agnostic to the way MEC is deployed.

Realization of the Data Plane

- **Option 2**: For performance enhancements, it can make sense to re-use the SFC functionality provided by the underlying NFVI for traffic routing. In such a deployment, the Data Plane as a dedicated component is not needed, and consequently, also the Mp2 reference point does not exist. The SFC functionality in the NFVI will be configured by the NFVO in the VIM based on the NFP of the NFV NS, using the Or-Vi reference point. The MEAO will need to translate the traffic rules into an NFP and send it to the NFVO. The ME platform will not control the traffic redirection directly via Mp2 will pass requests to activate / deactivate / update for traffic rules to the MEPM-V which will forward them to the MEAO. When receiving such a request, the MEAO will request the NFVO to update the NFP accordingly.

Key issue #1
Mapping of ME app VNFs to Network Services

- **Problem description**:
  - The ME applications appear like VNFs towards the ETSI NFV MANO components.
    - The Mobile Edge Orchestrator (MEO), as defined in the MEC reference architecture ETSI GS MEC 003, is transformed into a "Mobile Edge Application Orchestrator" (MEAO) that uses the NFVO for resource orchestration, and for orchestration of the set of ME app VNFs as one or more NFV NSs.
  - This is consistent with how NFV conceptualizes NFVO: as an entity that operates on NSs. For example, ETSI GS NFV-IFA 013, Annex A states as "Principle #3":
    - With respect to the Os-Ma-nfvo reference point, any interaction concerning a VNF is associated with at least one NS instance.

Key issue #1
Mapping of ME app VNFs to Network Services

- **Problem description**: (cont'd)
  - The issue is related to the mapping of ME app VNFs to NFV NSs. The MEAO would need to have a mapping from ME app VNFs to NSs.
  - The assumption is that Mv1 looks similar to Os-Ma-nfvo and that MEAO is creating the needed NSs for the ME app VNFs.
  - This assumption should be clearly stated in the document and the mapping of NS to ME app VNFs be explained. Functional blocks of the architecture request the creation/LCM of ME app VNF instances from the MEAO using the 2 reference points Mm1 and Mm9.
  - The Mm1 reference point between the MEO and the OSS is used for triggering the instantiation and the termination of ME app VNFs in the ME system.

Key issue #1
Mapping of ME app VNFs to Network Services

- **Problem description**: (cont'd)
  - The Mm9 reference point between the user application LCM proxy and the MEO of the ME system is used to manage ME applications requested by UE application.
  - None of those 2 reference points are dealing with services, while the MEAO would need NS knowledge, as well as the mapping of ME app VNF to NS.

Key issue #1
Mapping of ME app VNFs to Network Services

- **Solution(s)**:

  It is assumed that the MEAO would perform the following tasks:

  1) Request the NFVO to set one or more NSs to manage the ME app VNFs, which includes:
     - Generation and onboarding of an NSD
     - Requesting the instantiation of a NS
  2) When a new ME app VNF package is on boarded:
     - Updating of the NSD to reference the new package, and onboarding of the updated NSD

Key issue #1
Mapping of ME app VNFs to Network Services

- **Solution(s)**: (cont'd)

  3) When the instantiation of a ME app VNF is requested (see also key issue #13):
     - Request the NFVO to perform the UpdateNs operation, adding a new ME app VNF instance, and possibly, if the ME app VNF package was not included in the NSD on which the NS is built, updating the NS to use the new updated NSD
     - Once this was successful, trigger the MEPM-V to perform additional configuration
  4) When the termination of a ME app VNF is requested (see also key issue #14):
     - Request the NFVO to perform the UpdateNs operation, removing the ME app VNF instance

Key issue #1
Mapping of ME app VNFs to Network Services

- **Solution(s)**: (cont'd)

  The MEAO would need to track for every ME app VNF instance the following information:

  1) Identifier NS instance of which the ME app instance is part
  2) Identifier of the ME app VNF instance
  3) Responsible MEPM-V instance
  4) Responsible MEC Platform instance

  This information can be obtained for various sources:

  - 1) is available to the MEAO as part of managing the NSs
  - 2) is available via notifications on Mv1 regarding the result of the UpdateNs operation
  - 3) can be derived from 4)
  - 4) is configured into the ME app VNF instance upon instantiation, as per decision of the MEAO

Key issue #2
Usage of NFV Network Service

- **Problem description**:
  - NFV decouples software implementations of Network Functions from the computation, storage, and networking resources they use. The virtualization insulates the Network Functions from those resources through a virtualization layer. VNFs can be chained with other VNFs and/or Physical Network Functions (PNFs) to realize a NS.
  - As defined in the ETSI GS NFV-IFA 014, the NS describes the relationship between VNFs and possibly PNFs that it contains and the links needed to connect VNFs that are implemented in the NFVI network. Besides the VNF information, the NS information element also includes PNF information element, Virtual Link (VL) information element and VNF Forwarding Graph (VNFFG) information element. The VNFFG describes the topology of the NS by referencing VNFs and PNFs and VLs that connect them.

Key issue #2
Usage of NFV Network Service

- **Problem description**: (cont'd)
  - The Network Service Descriptor (NSD) can also be used to describe dependencies between VNFs, e.g. to ensure the ME Platform VNF is instantiated before any ME App VNF that uses it is instantiated.
  - In the present document, the concept of NS needs to be considered to deploy MEC in a NFV environment. Besides describing the dependencies between VNFs, it also defines the topology of the connections between the VNFs using the VNFFG.

Key issue #2
Usage of NFV Network Service

- **Solution(s)**:
  - It is recommended to introduce the NS concept in support of MEC deployment in NFV environment, while keeping the assumptions that both ME platform and ME applications are considered VNFs. The set of ME platform VNF and ME app VNFs can be grouped into one or more (possibly nested) NSs. In an example deployment, a NS could contain a ME platform VNF and one or more ME app VNFs, with which the following scenarios can be addressed:
  - the dependency between a ME app VNF and the ME platform VNF serving it are defined in the NSD;
  - the connectivity between ME app VNFs, and/or between ME app VNFs and the ME platform VNF, are defined in the VNFFG.

Key issue #2
Usage of NFV Network Service

- **Solution(s)**: (cont'd)
  - The service chaining aspects between multiple ME app VNFs are to be determined. It is further to be determined how to realize MEC in a NFV environment with multiple NFVI Points of Presence (PoPs), and how the service chaining works in such environment.
  - It is also to be determined whether or not additional functions would be needed for the NS defined in ETSI ISG NFV.

Key issue #3
Communication between MEAO and NFVO via Mv1

- **Problem description**:
  - The Mv1 reference point needs to allow the MEAO to invoke operations towards the NFVO to manage ME app VNFs. It also needs to allow the MEAO to onboard ME application packages into the NFVO. According to ETSI NFV standards, VNFs are always included in NSs

- **Solution(s)**:
  - The following set of interfaces and operations is defined for the Os-Ma-nfvo reference point (see ETSI GS NFV-IFA 013).

Interface	Operation	Direction	Remarks
NSD Management	On-board NSD	OSS → NFVO	On-board an NSD
	Update NSD	OSS → NFVO	Update an on-boarded NSD (e.g. to add reference to a new VNFD)
	Delete NSD	OSS → NFVO	Delete a NSD
	Query NSD	OSS → NFVO	Query list of NSDs
	(PNFD Management operations)	OSS → NFVO	Manage PNFDs, probably not relevant for MEC
	Notify	NFVO→ OSS	Asynchronous notifications on NSD change events
	Manage subscription	OSS → NFVO	Subscribe to notifications on NSD change events
NS LCM	Create NS Identifier	OSS → NFVO	Create a new NS identifier
	Instantiate NS	OSS → NFVO	Instantiate an NS
	Scale NS	OSS → NFVO	Scale an NS (i.e. change number of VNF instances)
	Update NS	OSS → NFVO	Update an existing NS (e.g. add VNF)
	Query NS	OSS → NFVO	Query list of NSs
	Terminate NS	OSS → NFVO	Terminate an NS
	Delete NS identifier	OSS → NFVO	Delete an NS identifier
	Heal NS	OSS → NFVO	Heal a malfunctioning NS
	Get Operation Status	OSS → NFVO	Track progress of NS LCM operations
	Notify	NFVO→ OSS	Asynchronous notifications on NSD LCM events
	Manage subscription	OSS → NFVO	Subscribe to notifications on NSD LCM events
NS PM	CreatePmJob	OSS → NFVO	Instruct the NFVO to create a job to collect performance information related to a NS
	QueryPmJob	OSS → NFVO	Instruct the NFVO to query PM jobs.
	DeletePmJobs	OSS → NFVO	Instruct the NFVO to delete PM jobs
	CreateThreshold	OSS → NFVO	Instruct the NFVO to create a threshold on a certain performance measure related to an NS. A threshold will trigger a notification if crossed.
	QueryThreshold	OSS → NFVO	Instruct the NFVO to query thresholds.
	DeleteThresholds	OSS → NFVO	Instruct the NFVO to delete thresholds.
	Notify	NFVO → OSS	Asynchronous notifications on PM events related to NSs (performance report available, threshold crossed)
	Manage subscription	OSS → NFVO	Subscribe to PM notifications
NS Fault Management	Get Alarm List	OSS → NFVO	Read the alarm list
	Notify	NFVO → OSS	Asynchronous notifications on alarms related to NSs
	Manage subscription	OSS → NFVO	Subscribe to notifications on alarms
VNF Package Management	On-board VNF Package	OSS → NFVO	On-board a VNF package into the NFVO
	Enable/Disable VNF Package	OSS → NFVO	Enable/disable an on-boarded VNF package
	Delete VNF Package	OSS → NFVO	Delete an onboarded VNF package
	Query On-boarded VNF Packages	OSS → NFVO	Obtain the list of on-boarded VNF packages base on certain criteria
	Fetch VNF package	OSS → NFVO	Obtain a copy of the VNF package content
	Fetch VNF Package Artifacts	OSS → NFVO	Obtain a copy of individual files from the VNF package
	Abort VNF Package Deletion	OSS → NFVO	Revoke a pending deletion request
	Notify	NFVO → OSS	Asynchronous notifications on events related to VNF Package changes
	Manage subscription	OSS → NFVO	Subscribe to notifications on events related to VNF Package changes

Key issue #4
Communication between VNFM and MEPM-V via Mv2

- **Problem description**:
  - In the MEC in NFV architecture, the MEPM is split into a MEC-specific part (MEPM-V) and a VNFM that is responsible for the LCM of the ME app VNF instances. Also, the orchestration flow is modified to take the NFVO into the loop (see key issue #3, clause 6.3). The MEPM-V needs to be able to communicate with the VNFM to keep track of LCM operations that were initiated by the NFVO. Also, as the MEPM-V has no longer access to the VIM, it may need to receive PM counters and alarms for the virtualized resources that are related to the ME app VNFs that are related to the ME platform instance which is managed by that MEPM-V.

Key issue #4
Communication between VNFM and MEPM-V via Mv2

- **Solution(s)**:
  - Mv2 is used for the communication between the VNFM that is responsible for the LCM of the ME app VNFs and the MEPM-V. In the ETSI NFV architecture, the reference point Ve-Vnfm-em (see ETSI GS NFV-IFA 008) is defined for the communication between the EM that manages the VNF instance and the VNFM that manages the VNF instance. In MEC, the MEPM-V is the EM of the ME platform, but also has certain management responsibilities also for the ME app instance (see ETSI GS MEC 010-1).
  - The following set of interfaces and operations is defined for the Ve-Vnfm-em reference point ETSI GS NFV-IFA 008.

Interface	Operation	Direction	Remarks
VNF LCM	CreateVnfIdentifier	EM → VNFM	Create an identifier for a new VNF instance
	InstantiateVnf	EM → VNFM	Instantiate a new VNF instance
	ScaleVnf	EM → VNFM	Scale a VNF instance
	ChangeVnfFlavour	EM → VNFM	Change VNF instance flavor
	TerminateVnf	EM → VNFM	Terminate a VNF instance
	DeleteVnfIdentifier	EM → VNFM	Delete a VNF identifier
	QueryVnf	EM → VNFM	Obtain VNF instance information
	HealVnf	EM → VNFM	Heal a VNF instance
	OperateVnf	EM → VNFM	Change the operational status of a VNF instance (started, stopped)
	ModifyVnfInformation	EM → VNFM	Modifies VNF information, including configurable properties.
	GetOperationStatus	EM → VNFM	Track progress of LCM operations
	Notify	VNFM → EM	Asynchronous notifications on LCM events
	Manage subscription	EM → VNFM	Subscribe to notifications on LCM events
VNF PM	CreatePmJob	EM → VNFM	Instruct the VNFM to create a job to collect virtualised resource performance information related to a (set of) VNF/VNFC instance(s)
	QueryPmJob	EM → VNFM	Instruct the VNFM to query PM jobs.
	DeletePmJobs	EM → VNFM	Instruct the VNFM to delete PM jobs
	CreateThreshold	EM → VNFM	Instruct the VNFM to create a threshold on a certain virtualised resource performance measure related to a (set of) VNF instance(s). A threshold will trigger a notification if crossed.
	QueryThreshold	EM → VNFM	Instruct the VNFM to query thresholds.
	DeleteThresholds	EM → VNFM	Instruct the VNFM to delete thresholds.
	Notify	VNFM → EM	Asynchronous notifications on PM events related to virtualised resource (performance report available, threshold crossed)
	Manage subscription	EM → VNFM	Subscribe to notifications on PM notifications
VNF Fault Management	Get Alarm List	EM → VNFM	Read the alarm list
	EscalatePerceivedSeverity	EM → VNFM	Propose an escalation of the severity of an alarm
	AcknowledgeAlarms	EM → VNFM	Acknowledge one or more alarms
	Notify	VNFM → EM	Asynchronous notifications on alarms related to virtualised resource
	Manage subscription	EM → VNFM	Subscribe to notifications on alarms
VNF Indicator	GetIndicatorValue	VNFM → EM	Read VNF-specific indicators
	Notify	EM → VNFM	Asynchronous notifications on Indicator event
	Manage subscription	VNFM → EM	Subscribe to notifications on Indicator changes

Key issue #5

Communication between VNFM and ME app instance via Mv3

- **Problem description**:
  - As defined in ETSI GS NFV-IFA 008, the Ve-Vnfm-vnf reference point allows the VNF instance to interact with the VNFM in various ways. The reference point is optional for the VNF instance to support. It is proposed by the present document to add to the MEC architecture a reference point Mv3 that is considered similar to Ve-Vnfm-vnf, to allow the VNFM and the managed ME app VNF instance to interact directly. This key issue highlights the possible interactions between these two entities using Ve-Vnfm-vnf.
- **Solution(s)**:
  - The following set of interfaces and operations is defined for the Ve-Vnfm-vnf reference point.

Interface	Operation	Direction	Remarks
VNF LCM	ScaleVnf	VNF → VNFM	"Scale Me" operation
	HealVnf	VNF → VNFM	"Heal" operation
	GetOperationStatus	VNF → VNFM	Track progress of scale and heal operations
	QueryVnf	VNF → VNFM	Obtain VNF instance information
	Notify	VNFM → VNF	Asynchronous notifications on LCM events
	Manage subscription	VNF → VNFM	Subscribe to notifications on LCM event
VNF PM	Notify	VNFM → VNF	Asynchronous notifications on PM events related to virtualised resource (performance report available, threshold crossed)
	Manage subscription	VNF → VNFM	Subscribe to notifications on PM notifications
VNF Fault Management	Get Alarm List	VNF → VNFM	Read the alarm list
	EscalatePerceivedSeverity	VNF → VNFM	Propose an escalation of the severity of an alarm
	AcknowledgeAlarms	VNF → VNFM	Acknowledge one or more alarms
	Notify	VNFM → VNF	Asynchronous notifications on alarms related to virtualised resource
	Manage subscription	VNF → VNFM	Subscribe to notifications on alarms
VNF Indicator	GetIndicatorValue	VNFM → VNF	Read VNF-specific indicators
	Notify	VNF → VNFM	Asynchronous notifications on Indicator event
	Manage subscription	VNFM → VNF	Subscribe to notifications on Indicator changes
VNF Configuration	SetConfiguration	VNFM → VNF	Set VNF configuration, e.g. during executing an LCM operation. The technology to be used on this interface is up to the VNF vendor to choose. ETSI GS NFV-SOL 002 [i.14] has defined a RESTful interface which is optional. ETSI GS NFV-IFA 008 [i.12] mentions cloud-init and HOT as possible technologies.

Key issue #6
AppD vs. VNFD for ME app VNFs

- **Problem description**:
  - It is assumed, in the present document, that the ME applications appear as VNFs so that ETSI NFV MANO functionality can be re-used. In addition, as specified in ETSI NFV, the VNF Descriptor (VNFD) is used in the VNF package management interface on the Or-Vnfm reference point. Information from the VNFD is used in VNF LCM procedures on the Or-Vnfm reference point. The same information is also used in the VNF LCM interface on the Ve-Vnfm reference point, but it is out of scope of ETSI NFV how this information is provisioned to the EM respectively the VNF.
  - From ETSI ISG MEC side, the application Descriptor (AppD) is defined in ETSI GS MEC 010-2. that is a part of application package and describes the application requirements and rules required by the application provider.
  - It needs to be analyzed how these two descriptors can work together; whether or not there is any impact on ME application package on-boarding and LCM of a ME application instance and if this is the case, what are the impacts.

Key issue #6
AppD vs. VNFD for ME app VNFs

- **Solution(s)**:
  - Mm3* reference point between MEAO and MEPM-V, which is based on the Mm3 reference point defined in ETSI GS MEC 003. Changes will be needed to this reference point to cater for the split between MEPM-V and VNFM (ME app VNF LCM). Based on an initial estimate, in a MEC in NFV deployment, Mm3* will not be involved in LCM specific procedures, as LCM is delegated to the VNFM. It is, therefore, assumed that Mm3* will be a subset of Mm3, with the LCM functionality not present, which means that no information from the VNFD will be exchanged on this reference point.

Key issue #6
AppD vs. VNFD for ME app VNFs

- **Solution(s)**: (cont'd)
  - Mv2 reference point between the VNFM of the ME app VNF with the MEPM-V. It is related to the Ve-Vnfm-em reference point as defined in ETSI NFV, but will possibly include additions, and might not use all functionality offered by Ve-Vnfm-em. It is assumed that the MEPM-V would receive Lifecycle Operation Occurrence Notifications w.r.t. the ME app VNF LCM procedures that were delegated to the VNFM. From that perspective, information from the VNFD could flow on that reference point.

Key issue #6
AppD vs. VNFD for ME app VNFs

- **Solution(s)**: (cont'd)
  - Mv3 reference point between the VNFM that manages a ME app VNF and the ME app VNF instance, which is related to the Ve-Vnfm-vnf reference point as defined in ETSI NFV. It is to be determined what additional features would be required. This optional reference point allows the VNF to invoke LCM operations w.r.t. itself from the VNFM, to send indicator information to the VNFM, and selected configurable VNF properties to be modified by the VNFM. Therefore, information from the VNFD would flow on this reference point, but it is out of scope of NFV how this information is provisioned to the VNF.

Key issue #6
AppD vs. VNFD for ME app VNFs

- **Solution(s)**: (cont'd)

  Two different options for the on-boarding:

  – In the first option, the package passes through the MEAO, which further on-boards the package into the NFV. The MEAO could just receive a ME app VNF package that contains both AppD and VNFD, and pass it on to the NFVO. It should be determined if AppD needs to be passed to the NFVO;

  – In the second option, the ME app VNF package is on-boarded like any other VNF package from the OSS into the NFVO, from where the MEAO can fetch it. This implies that the package on-boarded into the NFVO needs to contain both the VNFD and the AppD.

Key issue #7
VNF Package vs. MEC application package

- **Problem description**:
  - To represent the AppD (see clause 6.6), but also to provide other artifacts required by MEC management entities, there may be the need to provide additional MEC-specific data structures/files in the VNF package. It needs to be analyzed how MEC-specific files can be carried in the VNF package without interfering with the existing package content, and how these can be identified by the MEC management and orchestration components.

Key issue #7
VNF Package vs. MEC application package

- **Solution(s)**:

  A possible solution is to make use of a VNF package extension mechanism. The following requirements need to be met by such a mechanism:

  – It should allow to include files in the VNF package that are MEC-specific, without interfering with NFV-specific files, or file included based on other third party specifications.

  – It should allow a MEC management entity to identify the entry point to the set of MEC-specific files in the VNF package.

  – It should isolate the structure of the file set from other files in the package in a suitable way, allowing ETSI MEC to structure the file set in a way ETSI MEC sees necessary.

Key issue #8
VNF package onboarding

- **Problem description**:
  - When deploying MEC in the NFV environment, ME applications appear as VNFs towards the ETSI NFV MANO components. ETSI GS NFV-IFA 013 defines the on-boarding procedures of VNF packages into the NFVO. It is assumed that the VNF package (including the MEC specific descriptors) will be made available to the NFVO using the onboarding procedures in ETSI GS NFV-IFA 013. Two different deployment options are possible w.r.t. which component (NFVO or MEAO) is the master of the onboarding procedure, based on the existing mechanism defined in ETSI GS NFV-IFA 013.

Key issue #8
VNF package onboarding

- **Solution 1**:
  - If the MEAO is the master, the ME app package would first be provided by the OSS to the MEAO via Mm1, and onboarded to the NFVO by the MEAO via Mv1, using procedures defined in ETSI GS NFV-IFA 013. In that case, the MEC specific extensions of the VNF package are directly available to the MEAO, as the package passes through the MEAO.
- **Solution 2**:
  - If the NFVO is the master, the ME app package would be onboarded directly into the NFVO by the OSS via Os-Ma-nfvo. Via Mv1, the MEAO would be notified about package onboarding, and would be able to subsequently fetch whole packages or the needed package parts (so called package artifacts), using procedures defined in ETSI GS NFV-IFA 013 [i.6]. This would allow the MEAO to access the MEC specific extensions of the VNF package.

Key issue #9
Managing traffic redirection

- **Problem description**:
  - The Mp2 reference point is used by the ME platform to control the Data Plane (also known as forwarding plane) to configure the routing of traffic flows which will allow applications to originate, terminate or modify user plane traffic. The MEC requirements also include chaining multiple ME app VNFs to one chain through which the traffic is passed sequentially (see ETSI GS MEC 002, clause 6.2.4). ETSI MEC is considering the Mp2 reference point to be based on vendor-specific solutions, and has not specified any technical solution for it.
  - In the MEC architecture, traffic rules are received by the MEPM via Mm3 from the MEO, and passed via Mm5 towards the ME platform, which then configures the Data Plane based on these rules. Applications request the ME platform (via Mp1) to modify / activate / deactivate the traffic rules. Such changes would be communicated by the ME platform via Mp2 to the Data Plane.

Key issue #9
Managing traffic redirection

- **Solution 1**:
  - The Data Plane is realized as a PNF or VNF and exposes the Mp2 reference point towards the Mobile Edge Platform VNF.
- **Solution 2**:
  - NFV MANO can provide NFVI network resource management for the VNF to achieve certain traffic engineering without application level visibility, e.g. creating/updating NFP (Network Forward Path) rules. A NFP is defined as part of the VNFFG which is managed by the NFVO as part of the NS and can be modified via the Os-Manfvo reference point as defined in ETSI GS NFV-IFA 013. The NFVO can configure an NFP into the VIM via the Or-Vi interface as defined in ETSI GS NFV-IFA 006. The concept of the NFP has been introduced in ETSI GS NFV-MAN 001.

Key issue #9
Managing traffic redirection

- **Solution 3**:
  - The NFV architectural framework also enables a VNF to provide instructions to a dedicated switch on how to process the traffic. This is known as the DOPFR (Dynamic Optimization of Packet Flow Routing) feature as defined in ETSI GS NFV-IFA 018. The concept of DOPFR has been introduced in ETSI GS NFV-INF 005, clause 10.1..

Key issue #10
Comparison of AppD and VNFD data structures

- **Problem description**:
  - ETSI NFV has defined the VNFD (ETSI GS NFV-IFA 011), which has certain overlaps with the MEC AppD (ETSI GS MEC 010-2). The mapping of the two needs to be analyzed.

Key issue #10
Comparison of AppD and VNFD data structures

- **Solution(s)**:

VNFD attribute	AppD attribute
vnfdId	appDId
vnfProvider	appProvider
vnfProductName	appName
vnfSoftwareVersion	appSoftVersion
vnfdVersion	appDVersion
	mecVersion
vnfProductInfoName	appInfoName
vnfProductInfoDescription	appDescription
vnfmInfo	
localizationLanguage	
defaultLocalizationLanguage	
vdu	

VNFD attribute	AppD attribute
>swImageDescriptor	swImageDescriptor
virtualComputeDesc	virtualComputeDescriptor
virtualStorageDesc	virtualStorageDescriptor
intVirtualLinkDesc	
vnfExtCpd	appExtCpd
	appServiceRequired
	appServiceOptional
	appServiceProduced
	appFeatureRequired
	appFeatureOptional
	transportDependencies
	appTrafficRule
	appDNSRule
	appLatency
deploymentFlavour	
>vnfLcmOperationsConfiguration	
	terminateAppInstanceOpConfig
	changeAppInstanceStateOpConfig
configurableProperties	
modifiableAttributes	
lifeCycleManagementScript	
elementGroup	
vnfIndicator	
autoScale	

Key issue #11
NFV construct that corresponds to Mobile Edge Host

- **Problem description**:
  - In the MEC architecture, the ME host contains an instance of a NFVI and runs an instance of the ME platform. When virtualising this architecture, the concept of a ME host becomes obsolete.
  - In MEC, the concept of a ME host is used mainly in ME application instantiation and in mobility, as follows:
    - Prior to application instantiation, the MEO selects an appropriate ME host to instantiate the ME application, fulfilling constraints of the ME application (such as e.g. latency requirements).
    - If a UE is moving, the ME host that serves the UE may need to change to meet the constraints of the service e.g. w.r.t. latency. To compensate for changes incurred by UE mobility, either another ME app instance in another ME host serves the UE (session mobility), or the ME app that serves the UE is handed over to another ME host (application mobility, SmartRelocation).
  - With dissolving the ME host in a NFV deployment, it is still necessary to understand what will be the counterpart of a ME host in a NFV based deployment to fulfil the mobility use cases.

Key issue #11
NFV construct that corresponds to Mobile Edge Host

- **Solution(s)**:
  - ETSI ISG NFV has defined several constructs to structure a NFVI, such as NFVI-PoP (basically, a data center) and Zone (a set of co-located and well-connected physical resources which is a subset of a NFVI-PoP).

Key issue #12
ME App VNF Instance Relocation

- **Problem description**:
  - As part of the optional MEC feature "SmartRelocation", a ME application instance (VNF) needs to be re-located from one location to another, while preserving current state of the application instance..

- **Solution(s)**:
  - ETSI ISG NFV has defined VNF migration in ETSI GS NFV-MAN 001. However, the VNF LCM interface defined in ETSI GS NFV-IFA 007 does not offer any operation(s) enabling this use case. Further, the Os-Ma-nfvo reference point as defined in ETSI GS NFV-IFA 013 does not offer any operation(s) that allows requesting the NFVO to initiate VNF migration.

Key issue #13
Application instantiation

- **Problem description**:
  - The ME app VNF instantiation procedure includes two parts. The procedure is successful only when the two parts are executed successfully:
    1) The first part is to deploy the ME app VNF, which includes creation of the necessary virtualized resources and initial configuration of the ME app VNF.
    2) The second part is to send configuration to the ME platform by MEPM-V. According to ETSI GS MEC 010-2 [i.10], this configuration includes the traffic rules, DNS rules, the required and optional services, and services produced by the application instance, etc. The MEPM-V needs to be triggered to send the configuration.

Key issue #13
Application instantiation

- **Solution 1**:
  - MEAO calls NFVO to deploy the ME app VNF. MEAO listens to the NS change notification to learn if ME app VNF was created successfully in the related NS, and then sends a request to the MEPM-V that registers the created ME app VNF instance with the MEPM-V, and passes the necessary configuration parameters. MEPM-V then sends configuration to the ME platform..
- **Solution 2**:
  - MEPM-V acts as EM and subscribes to VNF lifecycle change notifications sent by the VNFM. MEAO calls NFVO to deploy the ME app VNF. After the ME app VNF is deployed, a notification will be sent to MEPM-V. This notification will trigger the MEPM-V to obtain the configuration from the MEAO and to send it to the ME platform. A notification should be sent to MEAO to indicate the configuration result.

For both solutions, the configuration information comes from OSS via Mm2 reference point and/or from application descriptor via the MEAO.

Key issue #14
Application instance termination

- **Problem description**:
  - The ME app VNF instance termination procedure includes two parts:
    1) The first part is to remove the configuration in ME platform and Data Plane.
    2) The second part is to terminate the ME app VNF.
  - The MEPM-V needs to be triggered to remove the configuration.

Key issue #14
Application instance termination

- **Solution 1**:
  - MEAO sends an application instance termination request to MEPM-V. If graceful termination is requested, MEPM-V first indicates the ME platform to give time to the ME app for application level termination, and wait until successful or timeout. MEPM-V then requests ME platform to remove the configuration. After having received the application instance termination response, the MEAO requests NFVO to terminate the ME app VNF. NFVO then requests VNFM to shut down the ME app VNF and release the resources..
- **Solution 2**:
  - MEPM-V acts as EM and subscribes to VNF lifecycle change notifications sent by the VNFM. MEAO calls NFVO to terminate the ME app VNF. In case of graceful termination, the VNFM first arranges to take the ME app VNF out of service by interacting with MEPM-V. MEPM-V needs to indicate to the ME platform to give time to the ME app for application level termination. Once this is successful, or after a timeout, the VNFM shuts down the ME app VNF and releases the resources. After the ME app VNF has been terminated, a notification will be sent to MEPM-V. The MEPM-V will request ME platform to remove the configuration after receiving the notification.

Recommendation

1) Allow the MEAO to map a ME app VNF instance to a NS (key issue #1).
2) Define how the dependency between a ME app VNF and the serving ME platform VNF are defined in the NSD, and define how the ME app VNFs and the ME platform VNF are connected using the VNFFG. The service chaining aspect of ME app VNFs is for further elaboration (key issue #2).
3) Define the re-use of Os-Ma-nfvo for communication between the MEAO and the NFVO, and determine how to signal placement constraints for ME app VNF instances to the NFVO (key issue #3).
4) Determine which subset of the Ve-Vnfm-em reference point is required to be used on Mv2. No additions to this reference point are foreseen, rather subsetting (key issue #4).

Recommendation (cont'd)

5) Determine which subset of the Ve-Vnfm-vnf reference point is required to be used on Mv3. Particular attention needs to be paid to the VNF Configuration interface in that reference point. Also, it needs to be determined if there is an influence of this on the generic MEC architecture. No additions to this reference point are foreseen, rather subsetting (key issue #5).
6) Compare VNFD and AppD data structures, and determine if and how certain attributes can be mapped to each other. Also, determine whether the AppD can provide the minimum information set required for a VNFD, and what information needs to be added if it cannot (key issues #6 and #10).
7) Determine how MEC-specific information can be included in a VNF package (key issues #6 and #7).

Recommendation (cont'd)

8) Determine how the ETSI NFV onboarding procedures can be re-used for MEC in NFV (key issues #6 and #8).
9) Determine which NFV construct corresponds to the ME host in the generic MEC architecture (key issue #11).
10) Support for ME app VNF instance relocation is currently not available in ETSI NFV, consider addition of VNF migration (key issue #12).
11) Define the flows of ME app VNF instantiation and termination when deploying ETSI MEC in a NFV environment (key issues #13 and #14).