

ETSI GS Multi-access Edge Computing

001: Terminology

002: Use Cases and Requirements

003: Framework and Reference Architecture

발표자: 명세인(sein@pel.smuc.ac.kr)

지도교수: 이종혁(jonghyouk@pel.smuc.ac.kr)

상명대학교 프로토콜공학연구실

목 차

- Introduction
- MEC Requirements
- MEC Framework
- MEC Reference Architecture
 - Reference Point

Introduction

- 001: Terminology(v1.1.1 2016-03)
 - Provides a glossary of terms related to **conceptual, architectural and functional elements**
- 002: Use Cases and Requirements(v2.1.1 2018-10)
 - Specifies the requirements of MEC with the aim of promoting interoperability and deployments
- 003: Framework and RA(v1.1.1 2016-03)
 - Provides a **framework and reference architecture**
 - Describes a **ME system** enabling **ME app** to run efficiently and seamlessly in a mobile network

002: Requirements

- Generic principles
 - Introduction
 - The following principles are important to understand the context of Multi-access Edge Computing
 - Mobility support
 - Most devices connected to a 3GPP network are moving around within the **mobile network**
 - Some MEC app(s) are state-independent
 - do not need to keep state information related to the UEs they are serving
 - Other MEC app(s) are specifically related to the user activity
 - maintain some **application-specific user-related information** that needs to be provided to the instance of that **application running on another MEC host**

002: Requirements

- Generic principles
- Mobility support
 - As a consequence of UE mobility, the MEC system needs to support the following
 - continuity of the service
 - mobility of application(VM)
 - mobility of application-specific user-related information
- Deployment independence
 - Different deployment scenarios need to be supported
 - performance, costs, scalability, operator preferred deployments, etc.
 - Deployment at the **radio node, an aggregation point, edge of the Core Network**(distributed data center, gateway)

002: Requirements

- Generic principles
 - Simple and controllable APIs
 - Developing APIs is as simple as possible and directly answering the needs of applications
 - Smart application location
 - some applications might have requirements in terms of latency
 - the conditions might evolve over time and require the MEC system to change the location of the application
 - UEs are moving from cell to cell
 - Different locations may have different “costs”
 - The MEC system needs to provide a system-wide lifecycle management of applications for MEC applications need to run “at the right place”

002: Requirements

- Generic principles
 - Application mobility to/from an external system
 - to support service continuity when the **user context** and/or **application instance** is relocated
 - the system shall be able to relocate a **ME app** running in an external cloud environment

002: Requirements

- Generic requirements
- Requirements on the framework(4/6)
 - **MEC system** should attempt to reuse the **NFV virtualisation infrastructure** and its **management functionality Application**
 - Shall be possible to enable the deployment of MEC applications on the same infrastructure as ETSI NFV-based VNFs
 - Shall be possible to deploy the MEC platform on MEC hosts in various locations of the fixed, mobile and wireless networks
 - **radio nodes, aggregation points, gateways, and in a distributed data centre at the edge of the Mobile Core Network**
 - Shall be possible to deploy the MEC platform, applications and services in a more centralized location
 - operator's or service provider's data centre

002: Requirements

- Generic requirements
- Requirements on the framework(6/6)
 - **Shall be possible to deploy** the MEC host in various stationary or moving nodes
 - The MEC system should provide capability to interact with the 5G core network
- Application Lifecycle(3/6)
 - The MEC host shall be available for the hosting of MEC applications
 - The MEC management shall support the instantiation of an application on a MEC host within the MEC system
 - The MEC management shall support the instantiation of an application on a MEC host when required by the operator
 - This may be in response to a request by an authorized third-party

002: Requirements

- Generic requirements
- Application Lifecycle(6/6)
 - The MEC management shall support the termination of a running application when required by the operator
 - This may be in response to a request by an authorized third-party
 - The MEC management shall be able to identify MEC application requires to run
 - features and MEC services
 - additional features and MEC services it can use if available
 - The MEC management shall be able to identify which features and MEC services are available on a particular MEC host

002: Requirements

- Generic requirements
 - Application Environment(5/5)
 - Shall be possible to deploy **MEC applications** on different **MEC hosts** in a seamless manner
 - without a specific adaptation to the application
 - The MEC management shall be able to verify the authenticity of a MEC application
 - The MEC management shall be able to verify the integrity of a MEC application
 - The MEC system shall support distributed edge cloud deployments
 - Horizontal: peer to peer connectivity of the application components
 - Vertical: hierarchical connectivity between different application components
 - Should be possible to host a MEC platform in cloud
 - resources are owned, operated and orchestrated by third party

002: Requirements

- Generic requirements
- Support of mobility(4/4)
 - The MEC system shall be able to **maintain connectivity** between a **UE** and an **application instance**
 - Especially, UE performs a handover to another cell associated with the same MEC host
 - The MEC system shall be able to **maintain connectivity** between a **UE** and an **application instance**
 - Especially, UE performs a handover to another cell not associated with the same MEC host
 - The MEC platform may use available **RNI** to optimize the mobility procedures required to support service continuity
 - The MEC platform may use available **core network information** to optimize the mobility procedures required to support service continuity

002: Requirements

- Use Cases
 - Consumer-oriented services
 - innovative services that generally benefit directly the end-user, i.e. the user using the UE
 - gaming, remote desktop applications, augmented and assisted reality, cognitive assistance, etc.
 - Operator and third party services
 - innovative services that take advantage of computing and storage facilities close to the edge of the operator's network
 - usually not directly benefiting the end-user, but can be operated in conjunction with third-party service companies
 - big data, security, safety, enterprise services, etc.

002: Requirements

- Use Cases
 - Network performance and QoE improvements
 - aimed at improving performance of the network(not new services)
 - either via application-specific or generic improvements
 - The user experience is generally improved

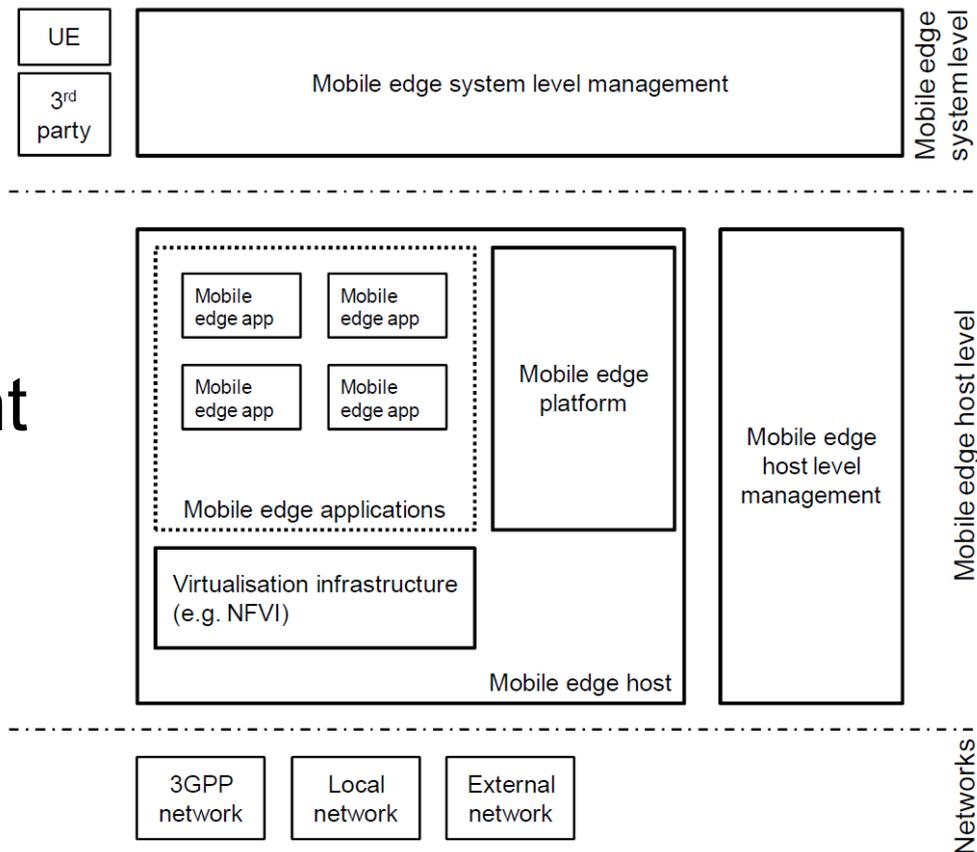
003: Framework

- Overview
 - The framework shows the structure of the **Mobile Edge Computing environment**
 - The RA shows the functional elements composing the **mobile edge system**
 - including the **mobile edge platform** and the **mobile edge management**, as well as the reference points between them
 - The functional elements and reference points describe the high-level functionality of the different functional elements and reference points
 - describes the high-level functionality of a number of **mobile edge services**

003: Framework

- MEC Computing framework

- ME host
 - ME platform
 - ME app
 - Virtualisation infrastructure
- ME system level management
- ME host level management
- External related entities
 - i.e. network level entities

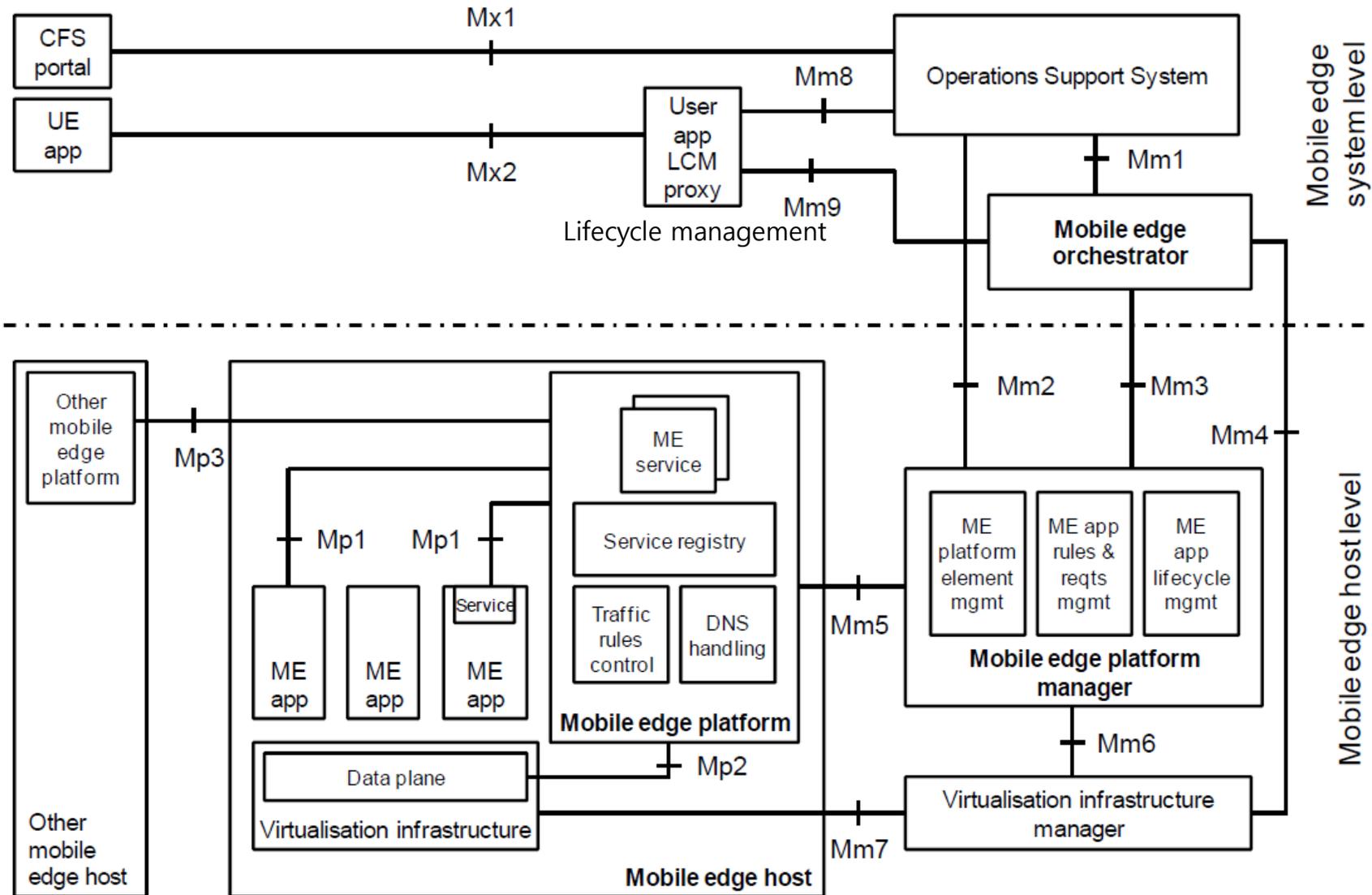


003: Reference Architecture

- Overview
- RA shows the functional elements composing the **ME system** and the **RP** between them
 - three groups of reference points defined between the system entities
 - RP regarding the mobile edge platform functionality (Mp)
 - management RP (Mm)
 - RP connecting to external entities (Mx)

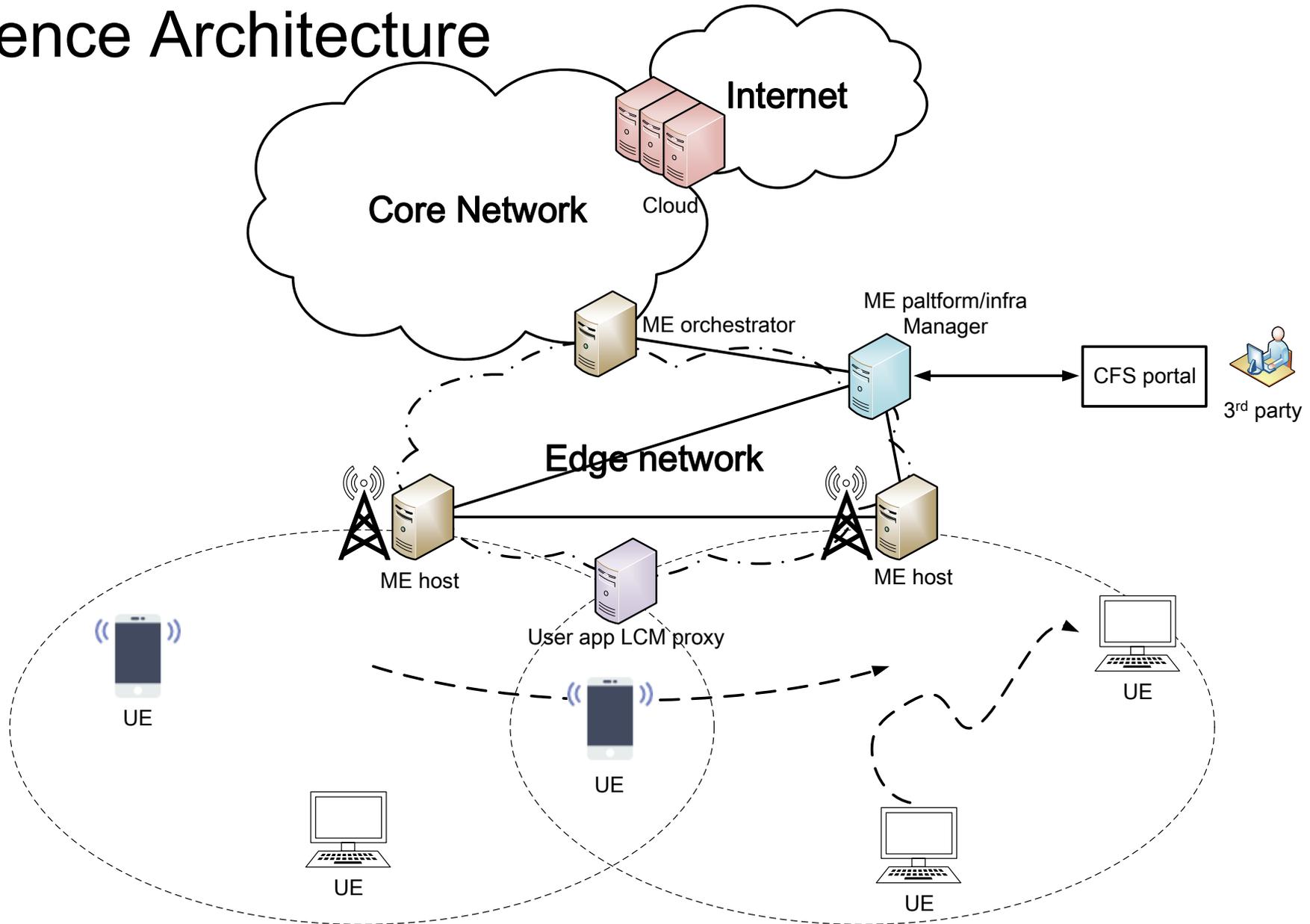
003: Reference Architecture

- Reference Architecture



003: Reference Architecture

- Reference Architecture



003: Reference Architecture

- **Functional elements**
 - **ME host**
 - an entity that contains the **ME platform** and a **virtualisation infrastructure**
 - provides compute, storage, and network resources for the ME app
 - **ME platform(responsible for the following functions)**
 - Offering an environment where the **ME app** can discover, advertise, consume and offer **ME services**
 - when supported, **ME services** available via **other platforms**
 - **Hosting ME services**
 - providing access to persistent **storage**
 - Receiving traffic rules from the **ME platform manager**, **app**, or **services**
 - Receiving DNS records from the **ME platform manager** and configuring a DNS proxy/server

003: Reference Architecture

- Functional elements
 - ME app
 - Running as **VM** on top of the virtualisation infrastructure provided by the **ME host**
 - Interact with the **ME platform** to consume and provide **ME services**
 - indicating availability, preparing relocation of user state
 - Rules and Requirements
 - Resources, max latency, required or useful services, etc.
 - Validated by the **ME system level management**

003: Reference Architecture

- Functional elements
 - ME system level management
 - ME orchestrator(Core functionality)
 - Maintaining an overall view of the **ME system** based on deployed **ME hosts**
 - available resources, available **ME services**, and topology
 - on-boarding of application packages
 - checking the integrity and authenticity of the packages
 - validating application rules and requirements
 - if necessary adjusting them to comply with operator policies
 - keeping a record of on-boarded packages
 - preparing the **virtualisation infrastructure manager(s)** to handle the applications
 - selecting appropriate **ME host(s)** for app instantiation based on constraints(latency, available resources, available services)
 - triggering application instantiation, termination, relocation(supported)

003: Reference Architecture

- Functional elements
 - ME system level management
 - Operations Support System(OSS)
 - receives requests via the **CFS portal** and from **UE app** for instantiation or termination of app(s), and decides on the granting of these requests
 - Granted requests are forwarded to the **ME orchestrator** for further processing
 - When supported, the OSS also receives requests from **UE app** for relocating app(s) between **external clouds** and the **ME system**

003: Reference Architecture

- Functional elements
 - ME system level management
 - User app lifecycle management proxy
 - A user application is a **ME app** instantiated in the **ME system**
 - response to a request of a user via an **UE app**
 - Allow **UE app** to request on-boarding, instantiation, termination of ME app and relocation(supported) in and out of the **ME system**
 - Also allow informing the **UE app** about the state of the **ME app**
 - Authorizes requests from **UE app** in the **UE** and interacts with the **OSS** and the **ME orchestrator** for further processing
 - Only accessible from within the Mobile Network
 - Only available when supported by the **ME system**

003: Reference Architecture

- Functional elements
 - ME host level management
 - ME platform manager
 - Managing the life cycle of app(s) including informing the **ME orchestrator** of relevant application related events
 - Providing element management functions to the **ME platform**
 - Managing the app rules and requirements
 - service authorizations, traffic rules, DNS configuration and resolving conflicts
 - also receives virtualised resources fault reports and performance measurements from the **virtualisation infrastructure manager** for further processing

003: Reference Architecture

- Functional elements
 - ME host level management
 - Virtualisation infrastructure manager
 - allocating, managing and releasing virtualized resources of the virtualisation infrastructure
 - compute, storage and networking
 - preparing the virtualisation infrastructure to run a software image
 - preparation includes configuring the infrastructure
 - receiving and storing the software image
 - when supported, rapid provisioning of applications
 - “Openstack++ for Cloudlet Deployments”
 - collecting and reporting performance and fault information about the virtualised resources
 - when supported, performing application relocation
 - from/to external cloud environments
 - “Adaptive VM Handoff Across Cloudlets”

003: Reference Architecture

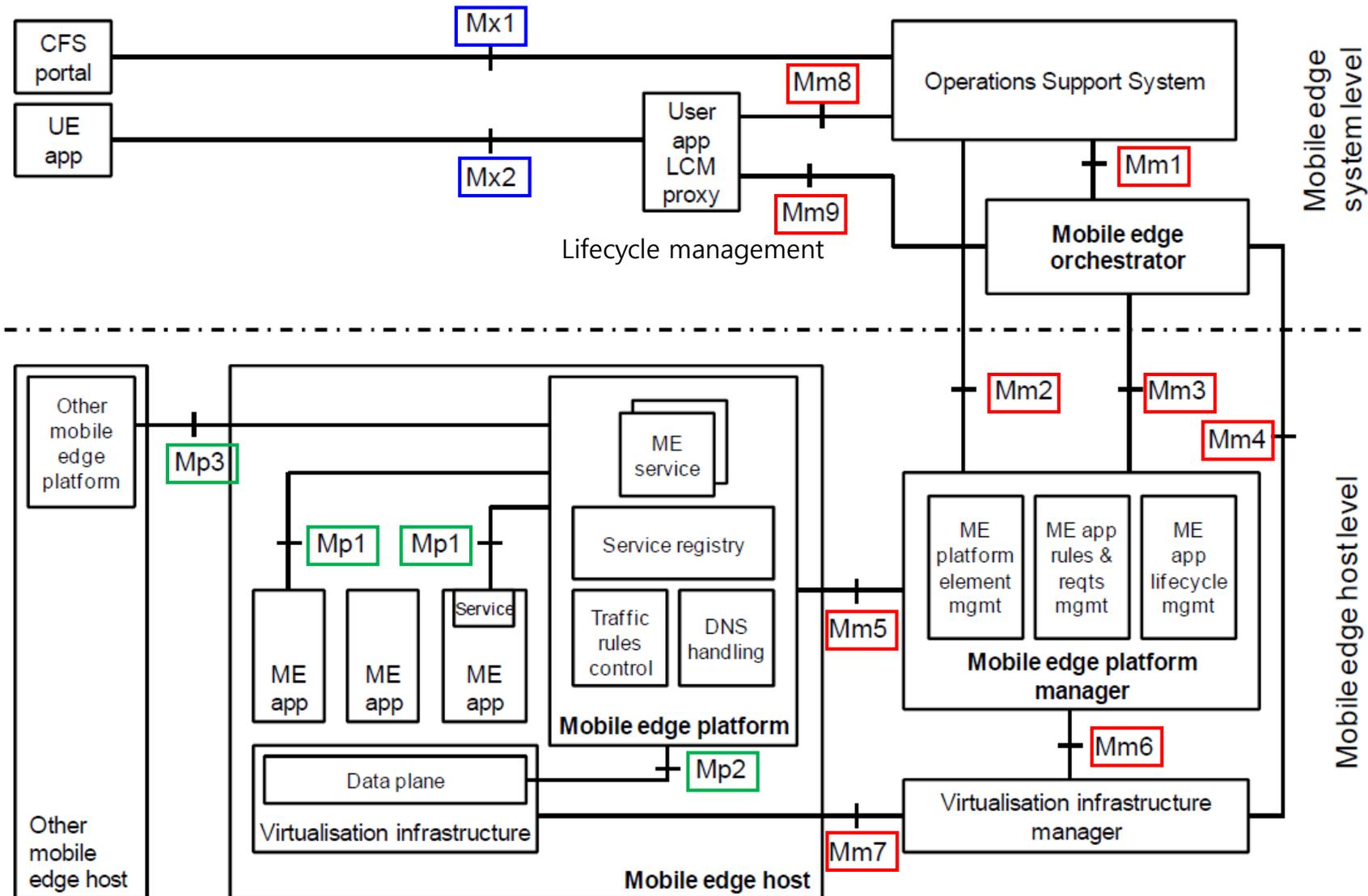
- Functional elements
 - UE app
 - As defined in the present specification, **UE app(s)** are applications in the **UE**
 - capability to interact with the **ME system** via a **user application lifecycle management proxy**
 - Customer facing service portal
 - allows operators' third-party customers to select and order a set of **ME app(s)** that meet their **particular needs**
 - receives back **service level information** from the provisioned applications.

003: Reference Architecture

- Reference Points
 - RP related to the **ME platform**
 - Mp1~3
 - RP related to the **ME management**
 - Mm1~9
 - RP related to **external entities**
 - Mx1~2

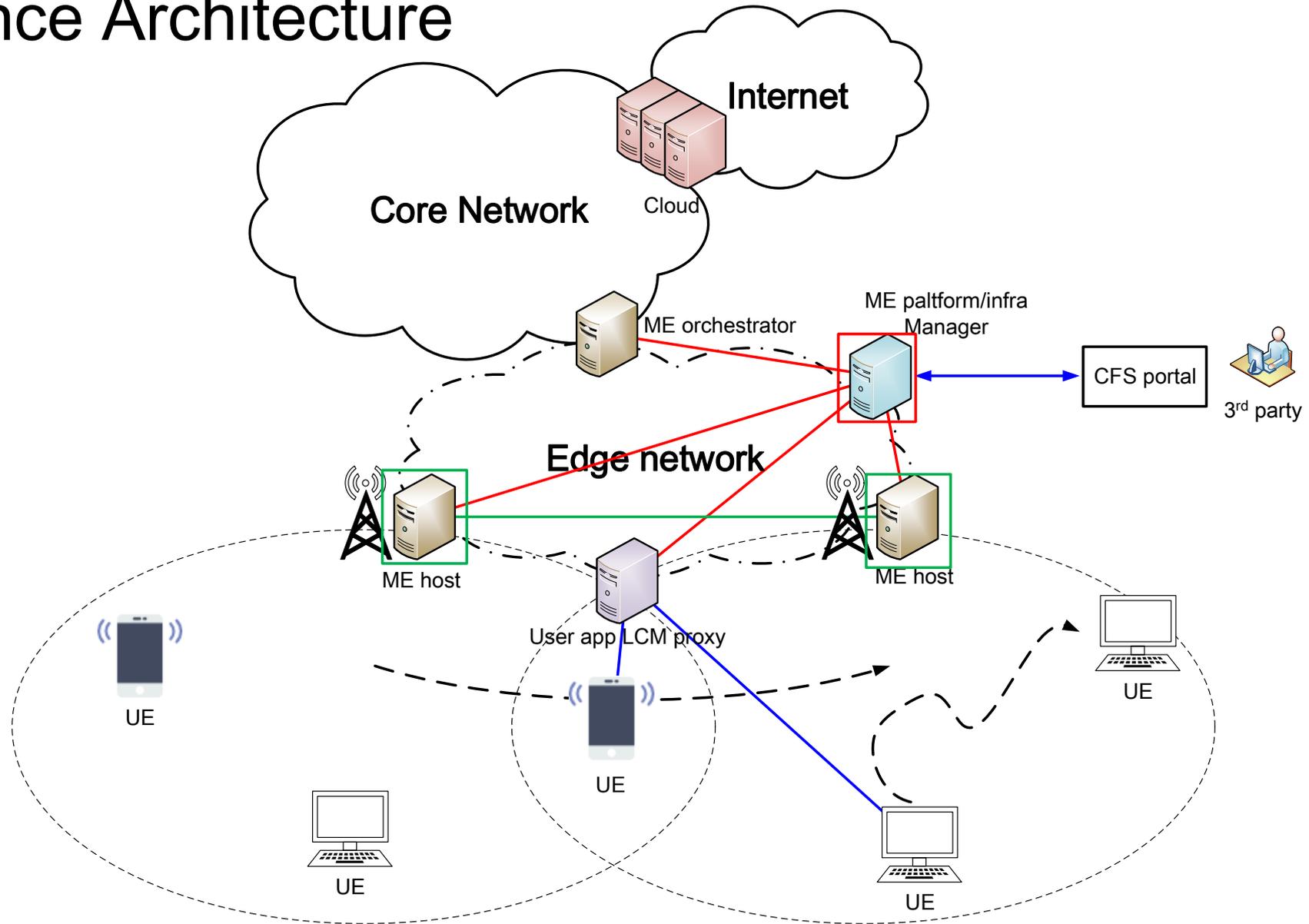
003: Reference Architecture

- Reference Architecture



003: Reference Architecture

- Reference Architecture



003: Reference Architecture

- Reference Points
 - RP related to the **ME platform**
 - Mp1
 - provides service registration, discovery and communication support
 - application availability, session state relocation support procedures, traffic rules and DNS rules activation, access to persistent storage and time of day information, etc.
 - Mp2(not further specified.)
 - instruct the data plane on how to route traffic among applications, networks, services, etc.
 - Mp3
 - control communication between **ME platforms**

003: Reference Architecture

- Reference Points
 - RP related to the **ME management**
 - Mm1
 - triggering the instantiation and the termination of ME apps in the ME system
 - Mm2
 - **ME platform configuration, fault and performance management**
 - Mm3
 - management of the application lifecycle, application rules and requirements
 - keeping track of available ME services
 - Mm4
 - manage virtualised resources of the **ME host**
 - keeping track of available resource capacity
 - managing **app images**

003: Reference Architecture

- Reference Points
 - RP related to the **ME management**
 - Mm5(not further specified)
 - perform platform configuration, configuration of the application rules and requirements, application lifecycle support procedures, management of application relocation, etc.
 - Mm6
 - manage virtualised resources
 - e.g. to realize the application lifecycle management
 - Mm7(not further specified)
 - manage the virtualisation infrastructure
 - Mm8(not further specified)
 - handle **UE app(s)** requests for running app(s) in the ME system
 - Mm9(not further specified)
 - manage **ME app(s)** requested by **UE app**

003: Reference Architecture

- Reference Points
 - RP related to **external entities**
 - Mx1(not further specified)
 - used by the third-parties to request the **ME system** to run applications in the **ME system**
 - Mx2
 - used by a UE application to request the **ME system** to run an application in the **ME system**
 - or to move an application in or out of the **ME system**
 - This reference point is only accessible within the **mobile network**
 - only available when supported by the **ME system**

003: Reference Architecture

- ME services
 - General
 - **ME service** is a service provided and consumed either by the ME platform or a ME app
 - When provided by an application, it can be registered in the list of services to the ME platform over the Mp1 reference point
 - Radio Network Information
 - RNI service, when available, provides authorized apps with radio network related information
 - appropriate up-to-date **RNI** regarding radio network conditions
 - measurement and statistics information related to the user plane
 - information related to UEs served by the radio node(s) associated with the **ME host**
 - (e.g. UE context and radio access bearers)
 - changes on information related to UEs served by the radio node(s) associated with the **ME host**

003: Reference Architecture

- ME services
 - Location
 - **when available**, provides authorized applications with location-related information
 - exposes information to applications, such as:
 - the location of specific UEs currently served by the **radio node(s)** associated with the **ME host**
 - information about the location of all UEs currently served by the **radio node(s)** associated with the **ME host**
 - optionally, information about the location of a certain category of UEs currently served by the **radio node(s)** associated with the **ME host**
 - a list of **UEs** in a particular location
 - information about the location of all radio nodes currently associated with the **ME host**

003: Reference Architecture

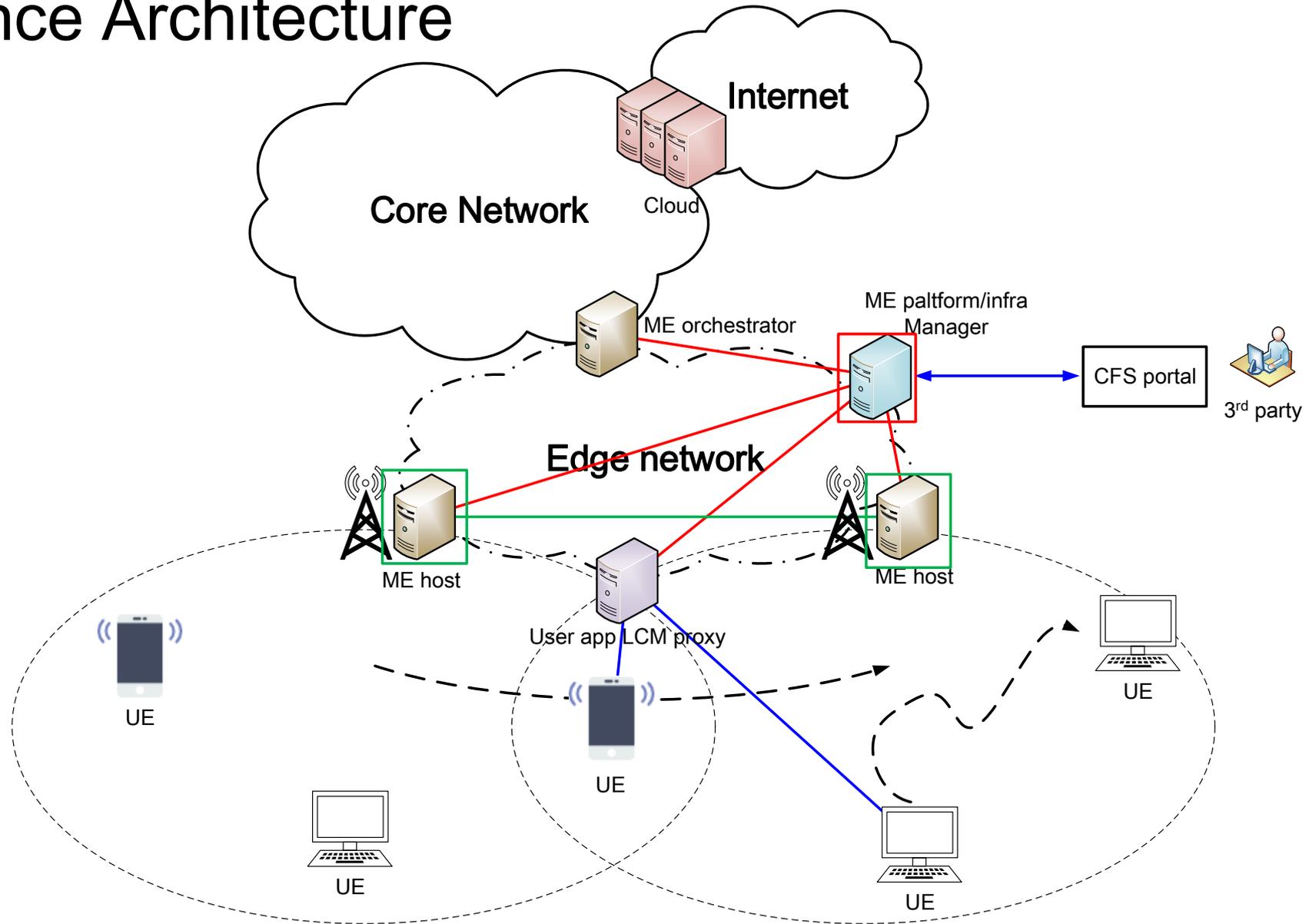
- ME services
 - Bandwidth Manager
 - when available, allows allocation of bandwidth to certain traffic routed to and from ME app(s) and the prioritization of certain traffic

003: Reference Architecture

- Support of application and UE mobility
 - **ME apps** not sensitive to **UE** mobility
 - **ME apps** sensitive to **UE** mobility
 - Maintaining connectivity between **UE** and **ME app** instance
 - Application state relocation
 - Application instance relocation within the **ME system**
 - Application instance relocation between the **ME system** and an **external cloud environment**

003: Reference Architecture

- Reference Architecture



감사합니다!