

**Question(s):** 7/2

Geneva, 17-26 September 2013

**TD****Source:** Editors**Title:** Requirements and analysis for management interface of virtualized resources in cloud computing (V0.2)**Summary**

This document specifies the requirements and analysis for the management interface between the cloud operational support system and the virtualized resource management [agent](#), ~~as well as the interface between the cloud operational support system and the virtualized resources~~. The functional requirements for the management interface are specified, which include configuration management, fault management, [and](#) performance management ~~and resource scheduling policy management~~. In the analysis part, the detailed information model supporting the above functions across the management interface is provided.

**Keywords**

Cloud computing, virtualized resource, Network Management Interfaces

**Introduction**

Virtualization is one of the features of cloud computing. In cloud computing environment, several virtualized resources share the physical infrastructures, thus provides a cost-effective and elastic way of the use of the resources. Compared with the traditional IT environment, besides the physical resources, the virtualized resources related to computing, storage and networking also need to be managed. It is important to provide standardized management interfaces for such virtualized resources in order to gain a better interoperability and help the large scale deployment and service provision of cloud computing.

~~Currently, there are some standardized works in this area, such as CIMI from DMTF, OCCI from OGF, and VRM API from IETF. The former two standardize the interface between the service consumer and the service provider, which acts as a service front-end to a provider's internal~~

**Contact:** WANG Ying  
MIIT/BUPT  
ChinaTel: +86-10-61198098 ext. 8514  
Fax: +86-10-62283119  
Email: [wangy@bupt.edu.cn](mailto:wangy@bupt.edu.cn)**Contact:** WANG Zhi-li  
MIIT/BUPT  
ChinaTel: +86-10-61198098 ext. 8726  
Fax: +86-10-62283119  
Email: [zlwang@bupt.edu.cn](mailto:zlwang@bupt.edu.cn)

**Attention:** This is not a publication made available to the public, but an **internal ITU-T Document** intended only for use by the Member States of ITU, by ITU-T Sector Members and Associates, and their respective staff and collaborators in their ITU related work. It shall not be made available to, and used by, any other persons or entities without the prior written consent of ITU-T.

management framework. And the latter one focuses on the interface between resource provider (hypervisor) and the resource manager.

This document is meant to standardize the interface between the cloud operational support system and the virtualized resource manager, as well as the interface between the cloud operational support system and the virtualized resources. The location of the interface is different from the previous works.

---

---

## Contents

1	Scope.....	5
2	References.....	6
3	Definitions .....	6
3.1	Terms defined elsewhere:.....	6
3.2	Terms defined in this Recommendation.....	6
4	Abbreviations and acronyms .....	7
5	Conventions .....	7
6	General overview .....	7
7.	Requirements .....	11
7.1	Business-level requirements.....	11
7.1.1	Requirements.....	11
7.1.2	Actor roles .....	16
7.1.3	Telecommunication resources .....	16
7.1.4	High-level use case diagrams .....	16
7.2	Specification-level requirements .....	21
7.2.1	Requirements.....	21
7.2.2	Actor roles .....	21
7.2.3	Telecommunications resources .....	21
7.2.4	Use cases .....	22
8	Analysis .....	39
8.1	Information object classes .....	39
8.1.1	Information entities imported and local label.....	39
8.1.2	Class diagram .....	39
8.1.3	Information object class definitions .....	41
8.1.4	Information relationship definitions.....	48
8.1.5	Information attribute definitions.....	52
8.2	Interface definition .....	55
8.2.1	Class diagram representing interfaces .....	55
8.2.2	Generic rules.....	56
8.2.3	ConfigurationManagement Interface (M) .....	56

8.2.4 FaultManagementOperations interface (M) .....	81
8.2.5 PerformanceManagementOperations interface (M) .....	81

## 1 Scope

This ~~draft Recommendation document~~ specifies the requirements and analysis for the management interface between the cloud operational support system (COSS) and the virtualized resource manager-management (VRM) agent ~~and the interface between the cloud operational support system and the virtualized resources~~. This ~~document~~ draft Recommendation follows the interface specification methodology described in [ITU-T M.3020].

In this ~~draft Recommendation document~~, the ~~virtualized resource management~~ VRM agent stands for the management and control aspects of abstraction and control functions within resource and network layer of the cloud computing reference architecture (CCRA) [ITU-T Y.CCRA]. The COSS represents an integrated management system across a global management domain which implements operational related management capabilities required in order to manage and control the cloud services offered to customers.

~~The virtualized resource management agent could be implemented as management software r is used to manage the virtualized resource cluster based on same or different virtualized technologies or just implemented as part of a hypervisor.~~

~~The virtualized resource management agent carries out the functionalities such as exercise control over, identify, collect data from and provide data to virtualized resources. One or more virtualized resource managers may be deployed depending on the virtualized technology, geographic distribution, or the scale of the virtualized resources. The cloud operational support system represents an integrated management system across a global management domain which implements operational related management capabilities required in order to manage and control the cloud services offered to customers. In order to exchange information about configuration, status, scheduling policies about virtualized resources to implement a centralized monitoring and administration of the co-existing virtualized resources to keep the cloud service operating normally, it is necessary to specify the management interface between virtualized resource managers or different virtualized technologies. The cloud operational support system and the communicates with virtualized resource managers or directly with virtualized resources (by hypervisor)ment agent through the management interface to realize its management functions.~~

[wzl1] In this draft Recommendation, ~~t~~The functional requirements for the management interface are specified, which include configuration management, fault management, and performance management ~~and strategy management~~. In the analysis part, the detailed ed information model supporting the above functions across the management interface is provided.

This draft Recommendation only focuses es on the computing and storage related virtualized resources ~~of in~~ cloud computing environment, such as resource pool, template, virtual machine, virtual machine image, volume, and network interface. Networking related virtualized resources such as virtualized network, virtualized link, virtualized node, and virtualized port can be derived from the corresponding Information Object Classes (MIOCs) defined in [ITU-T M.3160], which is out of the scope of this draft Recommendation.

## 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 M.3020] Recommendation ITU-T M.3020 (2011), *Management Interface Specification Methodology*
- [ITU-T M.3160] Recommendation ITU-T M.3160 (2008), *Generic, protocol-neutral management information model*
- [ITU-T M.3700] Recommendation ITU-T M.3700(2010), *Common management services – Object management – Protocol neutral requirements and analysis*
- [ITU-T M.3703] Recommendation ITU-T M.3703 (2010), *Common management services – Alarm management – Protocol neutral requirements and analysis*
- [ITU-T M.3704] Recommendation ITU-T M.3704 (2010), *Common management services – Performance management – Protocol neutral requirements and analysis*
- [ITU-T Y.CCRA] Draft Recommendation ITU-T Y.CCRA(2012<sup>3</sup>), *Cloud Computing Reference Architecture*
- [ITU-T Y.CCDEF] Draft Recommendation ITU-T Y.CCDEF(2013).

## 3 Definitions

### 3.1 Terms defined elsewhere:

- 3.1.1 ~~This Recommendation uses no terms defined elsewhere.~~<sup>[wzl2]</sup> cloud service provider [ITU-T Y.ccdef]
- 3.1.2 cloud service [ITU-T Y.ccdef]
- 3.1.3 component [ITU-T Y.ccra]
- 3.1.4 Information Object Class (IOC) [ITU-T M.3020]

Editor's note : check [ITU-T Y.ccra], or [ITU-T Y.ccdef] to add more terms used in this document

### 3.2 Terms defined in this Recommendation

~~This Recommendation has no new definitions.~~ Virtualized resource management agent<sup>[wzl3]</sup> (VRM Agent)

## 4 Abbreviations and acronyms

This Recommendation uses the following abbreviations and acronyms:

[CCRA](#) [Cloud computing Reference Architecture](#)

~~COSS~~ ~~Cloud~~ Operational Support System

~~CSP~~ ~~Cloud Service Provider~~

~~CSU~~ ~~Cloud Service User~~

IOC Information Object Class

IRP Integration Reference Point

RPL Resource Pool

RSP Resource Scheduling Policy

VM Virtual Machine

VR Virtualized Resources

VRM Virtualized Resources ~~Manager~~ [Management](#)

## 5 Conventions

This Recommendation uses [the conventions defined in \[ITU-T M.3020\]](#) for requirements capture and analysis.

## 6 General overview

In [Recommendation \[ITU-T ~~draft recommendation~~-Y.CCRA\]](#), the cloud computing reference architecture is specified, which includes five layers: User Layer, Access Layer, Service Layer, Resource & Network Layer, and Cross-Layer Functions. [The User Layer supports interaction between various roles and the underlying cloud architecture functional layers.](#) ~~[wzl4]The User Layer performs interaction between the CSU (Cloud Service User) and the cloud infrastructure. The Access Layer provides a common interface for both manual and automated access to the capabilities available in the Services Layer. The Access Layer provides a common interface for both manual and automated cloud service capabilities and service consumption. The Services Layer contains the implementation of the services provided by a cloud service provider. The Services Layer is where the CSP (Cloud Service Provider) orchestrates and exposes cloud services. The Resources and Network Layer is where the physical resources reside. This includes equipment typically used in a data centre, and also the corresponding non-cloud-specific software that runs on the servers and other equipment such as host operating systems, hypervisors, device drivers, generic systems management software. The Resources and Network Layer also represents and houses the cloud transport network functionality which is required to provide underlying network connectivity between the cloud service provider and the users, as well as within the cloud service provider and between peer service providers.~~ ~~[wzl5]~~ [The Resources and Network Layer components include Resource Abstraction & Control components and Physical Resources components. The Resource](#)

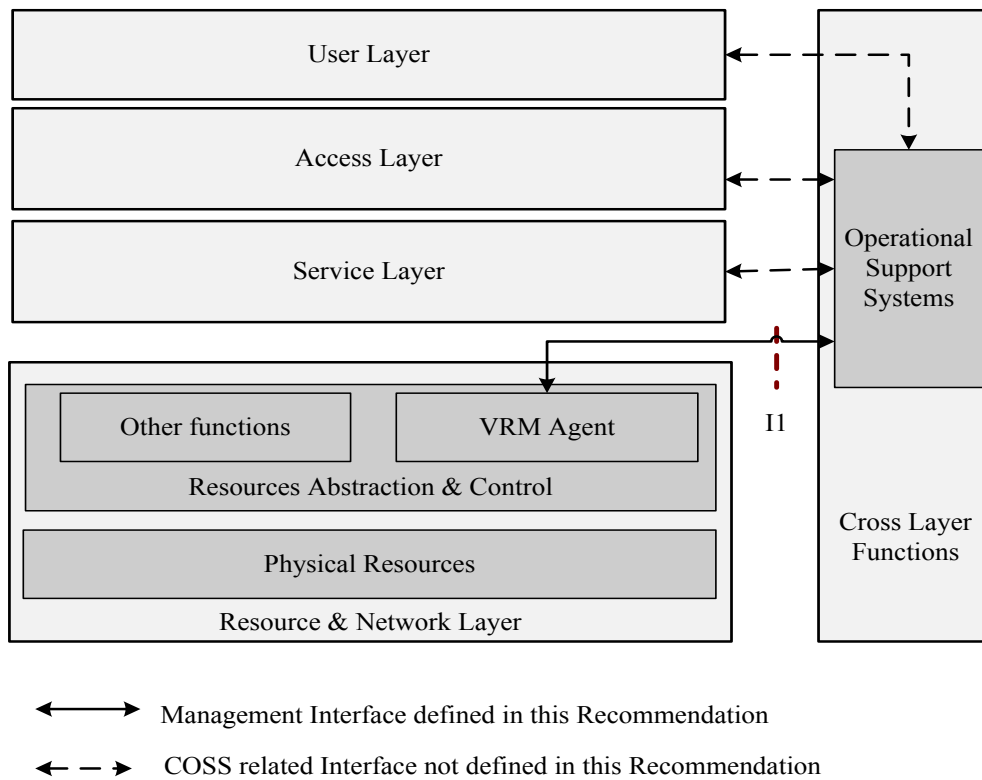
Abstraction & Control component is used by cloud service providers to provide access to the physical computing resources through software abstraction, and it also enables control functionality, enabling monitoring and management capabilities implemented in the Operational Support Systems component. ~~The Resources and Network layer is where the physical resources reside. And it is also responsible of management, monitoring, and scheduling of computing, storage, and network resources into consumable services, as well as pooling and virtualization of physical resources.~~ Cross-layer functions involve development ~~support~~function components, integration components, security components, operational support systems components and business support systems components.

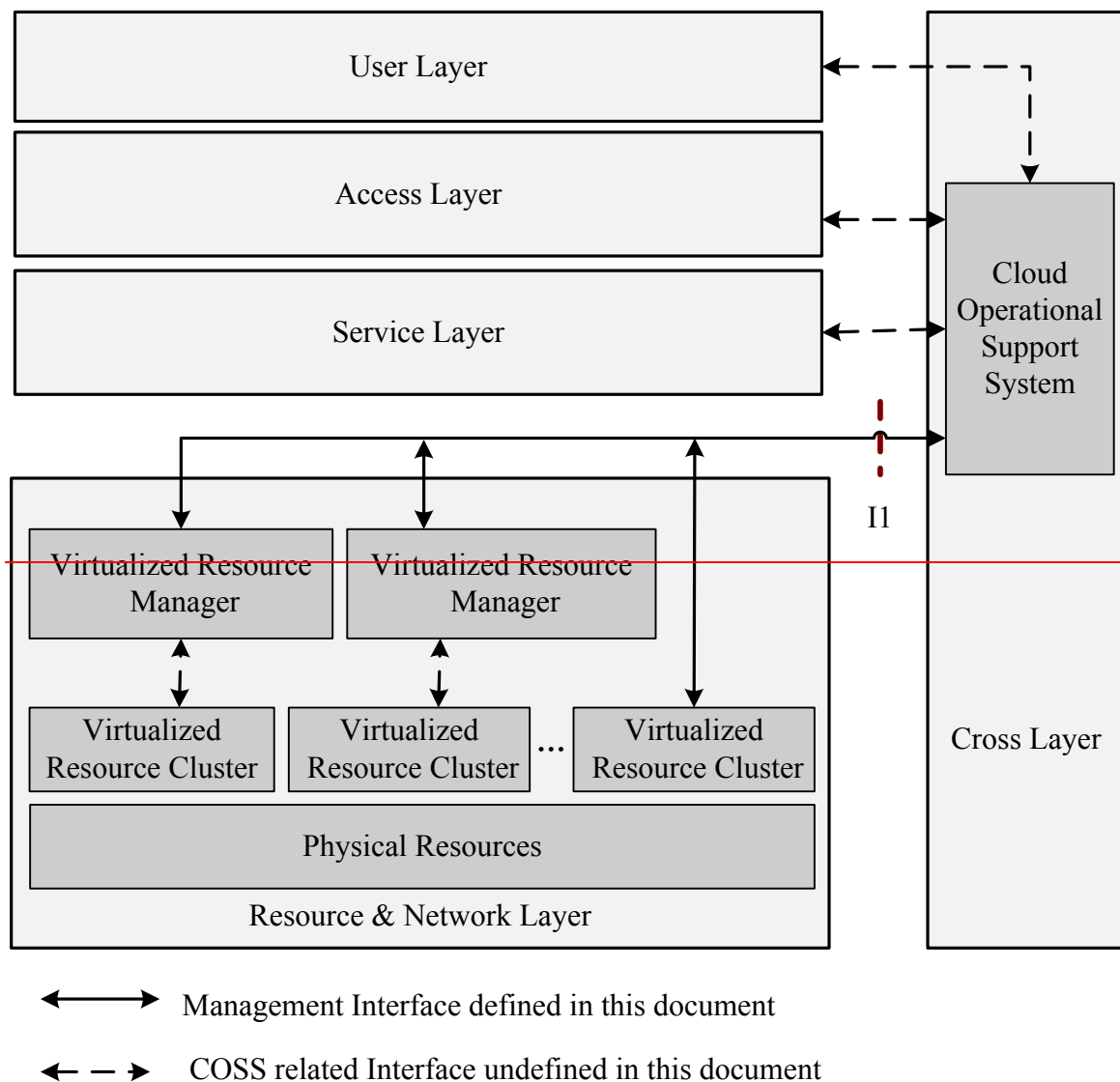
This ~~document draft Recommendation dwells deals with~~on the interface between operational support systems components management function of the Cross-layer function and the ~~virtualized resources management~~VRM agent within the Resources and Network layer. The location of the interface specified in this draft Recommendation within the cloud reference architecture CCRA of [ITU-T Y.CCRA] is illustrated as I1 in Figure 1 as I1.

~~In current cloud computing environment, the virtualized resources (VR) of a datacenter may consist of clusters of virtualized resources implemented through different system virtualized technologies, such as XEN, VMware, KVM, and so on. Each virtualized resource in a cluster may be managed by a virtualized resource manager (VRM) or be managed directly by a cloud operational support system (COSS). The VRM could be technology specific provided by virtualized technology suppliers. Or it could be a manager across different virtualized technologies. In the former case, the COSS together with the supporting management interface implement the unified management of the co-existing virtualized resources based on multiple virtualized technologies and at the same time conserve the technological features and flexibility between a technology specific VRM and its VRs. In the latter case, the COSS together with the supporting management interface could still implement a centralized monitoring and administration of the co-existing virtualized resources to keep the cloud system operating normally. Thus in both cases it is necessary to specify the interface between COSS and VRM, or COSS and VR.~~

~~The location of the interface specified in this document within the cloud reference architecture specified in Y.CCRA is illustrated in Figure1 as I1.~~







**Figure 1 – management interface of ~~cloud-computing~~ virtualized resources of cloud computing within the Cloud-computing-reference-architecture CCRA**

Where, the ~~Cloud-Operational-Support-Systems~~ cloud OSS components (~~COSS~~) encompass the set of operational related management capabilities required in order to manage and control the cloud services offered to customers. As the focus of this draft Recommendation is the management interface for virtualized resources, the ~~virtualized-resource-management~~ VRM agent (~~VRMA~~) is defined as an extension of Resource Abstraction & Control component within ~~r~~Resource and ~~n~~Network ~~l~~Layer of CCRA [ITU-T Y.CCRA] to stand for the management and control aspects of Resource ~~a~~Abstraction and ~~e~~Control functions. VRM agent ~~A~~ carries out the functionalities such as exercise control over, identify, collect data from and provide data to virtualized resources. The ~~virtualized-resource-management~~ VRM agent ~~could~~ may be implemented as management software used to manage a group of virtualized resources ~~clusters~~ based on the same or different virtualization ~~ed~~ technologies or just implemented as part of a hypervisor.

Through the management interface ~~With interaction with~~ VRM agent ~~A through a management interface~~, cloud ~~C~~ OSS ~~could~~ can ~~implement~~ achieve ~~a~~ the centralized ~~monitoring and administration~~ management of ~~the co-existing~~ virtualized resources in a multi-vendor

~~environment to keep the cloud service operating normally.~~

## 7. Requirements

### 7.1 Business-level requirements

#### 7.1.1 Requirements

This Recommendation focuses on the interface between the cloud operational support system and the virtualized resource manager or the interface between the cloud operational support system and the virtualized resources. Through the interface, COSS<sub>[wzl6]</sub> can provision and query the information of virtualized resources. And the VRM<sub>[wzl7]</sub> or VR can report the configuration changes, state changes, performance data, and fault event information of the virtualized resources to COSS.

The management requirements can be grouped into one of the following categories:

- Configuration management, involves the functions to exercise control over, identify, collect data from and provide data to computing and storage related virtualized resources such as resource pool, template, virtual machine, virtual machine image, network interface, resource scheduling policy ~~—(such as load balance, energy saving and so on)—~~, and file transfer, such as acquire available files, download available files, file download indication, notify file transfer ready and notify file preparation error, etc. <sub>[WY8]</sub>
- Fault management, involves reporting, filtering and synchronization of alarm .
- Performance management, involves the performance measurement management, query performance history data, performance threshold management, etc. ~~—~~

##### 7.1.1.1 Configuration management

Configuration management involves the functions to exercise control over, identify, collect data from and provide data to computing and storage related virtualized resources such as resource pool, template, virtual machine, virtual machine image, physical network interface, and virtual network interface. Such configuration management functions at the management interface consist of basic object management functions and some other specific management functions.

Editor's notes: add the sentence such as "in order to achieve the configuration management objectives, resource scheduling policy management and file transfer management functions are also needed".

The requirements for basic object management, including retrieve object information, create object, delete object, modify object attribute, generate attribute value change notification, generate state change notification, generate object create notification, generate object delete notification. This Recommendation reuses the requirements for basic object management specified in [ITU-T Rec. M.3700] and specifies the requirements beyond the basic object management functions,

Strictly speaking, the physical network interface is not a virtualized resource. However, it is highly related to the virtualized resources such as Virtual machine, and the management of virtual machine always requires corresponding management of physical network interface. Thus the physical

network interface is treated as an external resource, and it could be managed by other managers out of the scope of this Recommendation.

#### **7.1.1.1.1 Resource Pool Management**

- REQ-CC-FUN-01      The COSS should be able to create a resource pool through the management interface. The requirement REQ-OM-FUN-07 from [ITU-T Rec. M.3700] is reused here.
- REQ-CC-FUN-02      The COSS should be able to modify some attribute values of resource pool through the management interface. The requirement REQ-OM-FUN-09 from [ITU-T Rec. M.3700] is reused here.
- REQ-CC-FUN-03      The COSS should be able to delete a resource pool through the management interface. The requirement REQ-OM-FUN-08 from [ITU-T Rec. M.3700] is reused here.
- REQ-CC-FUN-04      The COSS should be able to get the information of a specific resource pool through the management interface. The requirement REQ-OM-FUN-03 from [ITU-T Rec. M.3700] is reused here.
- REQ-CC-FUN-05      The COSS should be able to get the information of the managed resources within a specified resource pool through the management interface. The requirement REQ-OM-FUN-04 from [ITU-T Rec. M.3700] is reused here

#### **7.1.1.1.2 Template management**

- REQ-CC-FUN-06      The COSS should be able to create a template through the management interface. The requirement REQ-OM-FUN-07 from [ITU-T Rec. M.3700] is reused here.
- REQ-CC-FUN-07      The COSS should be able to modify some attribute values of template through the management interface. The requirement REQ-OM-FUN-09 from [ITU-T Rec. M.3700] is reused here.
- REQ-CC-FUN-08      The COSS should be able to query the information of template through the management interface. The requirement REQ-OM-FUN-03 from [ITU-T Rec. M.3700] is reused here.
- REQ-CC-FUN-09      The COSS should be able to delete a template through the management interface. The requirement REQ-OM-FUN-08 from [ITU-T Rec. M.3700] is reused here.
- REQ-CC-FUN-10      The COSS should be able to acquire the templates which are maintained in VRM through the management interface.
- REQ-CC-FUN-11      The VRM should be able to download templates from COSS through the management interface.

#### **7.1.1.1.3 Virtual Machine management**

- REQ-CC-FUN-12      The COSS should be able to create a VM through the management interface. The requirement REQ-OM-FUN-07 from [ITU-T Rec. M.3700] is reused here.
- REQ-CC-FUN-13      The COSS should be able to modify some attribute values of VM through the management interface. The requirement REQ-OM-FUN-09 from [ITU-T Rec. M.3700] is reused here.

REQ-CC-FUN-14	The COSS should be able to query the information of VM through the management interface. The requirement REQ-OM-FUN-03 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-15	The COSS should be able to delete a VM through the management interface. The requirement REQ-OM-FUN-08 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-16	The COSS should be able to start a VM through the management interface.
REQ-CC-FUN-17	The COSS should be able to suspend a VM through the management interface.
REQ-CC-FUN-18	The COSS should be able to restore a VM which has been suspended through the management interface.
REQ-CC-FUN-19	The COSS should be able to shutdown a VM through the management interface.
REQ-CC-FUN-20	The COSS should be able to restart a VM which is in stopped state through the management interface.
REQ-CC-FUN-21	The COSS should be able to save a VM through the management interface.
REQ-CC-FUN-22	The COSS should be able to copy a VM through the management interface.
REQ-CC-FUN-23	The COSS should be able to migrate a VM from one physical server to another physical server through the management interface.

#### **7.1.1.1.4 Virtual Machine Image Management**

REQ-CC-FUN-24	The COSS should be able to create a virtual machine image through the management interface. The requirement REQ-OM-FUN-07 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-25	The COSS should be able to delete a virtual machine image through the management interface. The requirement REQ-OM-FUN-08 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-26	The COSS should be able to query the information of virtual machine images of specific VRMs through the management interface. And the VRM should also be able to query the information of virtual machine images of COSS through the management interface. The requirement REQ-OM-FUN-03 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-27	The COSS should be able to acquire the virtual machine image which is maintained in VRM through the management interface.
REQ-CC-FUN-28	The VRM should be able to download a virtual machine image from COSS through the management interface.

#### **7.1.1.1.5 Volume Management**

REQ-CC-FUN-29	The COSS should be able to create a volume through the management interface. The requirement REQ-OM-FUN-07 from [ITU-T Rec. M.3700] is reused here.
---------------	---

REQ-CC-FUN-30	The COSS should be able to delete a volume through the management interface. The requirement REQ-OM-FUN-08 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-31	The COSS should be able to modify a volume through the management interface. The requirement REQ-OM-FUN-09 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-32	The COSS should be able to query the information of volume through the management interface. The requirement REQ-OM-FUN-03 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-33	The COSS should be able to add a volume to a VM through the management interface.
REQ-CC-FUN-34	The COSS should be able to remove a volume from a VM through the management interface.

#### **7.1.1.1.6 Physical Network Interface Management**

REQ-CC-FUN-35	The COSS should be able to add a physical network interface through the management interface. The requirement REQ-OM-FUN-07 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-36	The COSS should be able to modify some attribute values of physical network interface through the management interface. The requirement REQ-OM-FUN-09 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-37	The COSS should be able to query the information of physical network interface through the management interface. The requirement REQ-OM-FUN-03 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-38	The COSS should be able to remove a physical network interface through the management interface. The requirement REQ-OM-FUN-08 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-39	The COSS should be able to activate a physical network interface through the management interface.
REQ-CC-FUN-40	The COSS should be able to deactivate a physical network interface through the management interface.

#### **7.1.1.1.7 Virtual Network Interface Management**

REQ-CC-FUN-41	The COSS should be able to create a virtual network interface for a VM through the management interface. The requirement REQ-OM-FUN-07 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-42	The COSS should be able to modify some attribute values of virtual network interface through the management interface. The requirement REQ-OM-FUN-09 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-43	The COSS should be able to query the information of virtual network interface through the management interface. The requirement REQ-OM-FUN-03 from [ITU-T Rec. M.3700] is reused here.
REQ-CC-FUN-44	The COSS should be able to delete a virtual network interface through the management interface. The requirement REQ-OM-FUN-08 from [ITU-T

Rec. M.3700] is reused here.

- REQ-CC-FUN-45      The COSS should be able to activate a network interface for some specified VM through the management interface.
- REQ-CC-FUN-46      The COSS should be able to deactivate a network interface for some specified VM through the management interface.

#### **7.1.1.1.8 Resource scheduling Policy Management**

- REQ-CC-FUN-47      The COSS should be able to dispatch a resource scheduling policy to VRM through the management interface.
- REQ-CC-FUN-48      The COSS should be able to query the resource scheduling policies of a VRM through the management interface.

**Editor's notes: as mentioned at the interim meeting, functions such as download, control and activation of policies should be considered and added if necessary.**

#### **7.1.1.1.9 File transfer management**

- REQ-CC-FUN-49      The COSS **could** [WY10] acquire the available file(s) stored in the VRM, such as template file(s) or virtual machine image file(s) and so on. The COSS should be able to send a request to the VRM, and on receiving this request, the VRM will prepare the file(s). When the files are ready successfully, the COSS will be notified and will then retrieve those files using some file transfer mechanisms, e.g., FTP. When the files are ready unsuccessfully, the COSS will be also notified and the error notification should indicate the reason.
- REQ-CC-FUN-50      The VRM **could** download the available file(s) stored in the COSS, such as template file(s) or virtual machine image file(s) and so on.  
The VRM should be able to send a request to the COSS to download file(s), and on receiving this request, the COSS will prepare the file(s). When the files are ready, the COSS will indicate the VRM to download the file(s), and then the VRM will retrieve those files using file transfer mechanisms, e.g., FTP.

#### **7.1.1.2 Fault management**

This Recommendation reuses the requirements specified in [ITU-T Rec. M.3703]..

#### **7.1.1.3 Performance management**

This Recommendation reuses the requirements specified in [ITU-T Rec. M.3704].

The performance parameters of the virtual resource in the cloud computing environment are as following:

- performance parameters of the VM: CPU utilization, memory utilization and disk utilization.
- performance parameters of the volume: storage utilization and I/O traffic.
- performance parameters of the network interface: **network traffic**, bandwidth, packet loss and time delay.

### 7.1.2 Actor roles

The capabilities described in this Recommendation are available and relevant to cloud operational support system and virtualized resource manager or the virtualized resource.

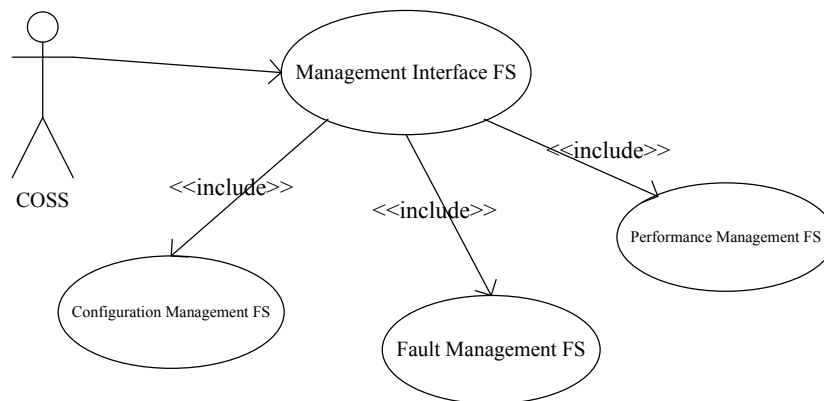
### 7.1.3 Telecommunication resources

The configuration, fault, performance and **resource scheduling policy** management functionality is applicable to the computing and storage related virtualized resources in cloud computing environment.

### 7.1.4 High-level use case diagrams

This clause contains high-level UML use case diagrams that summarize the functionality and interfaces.

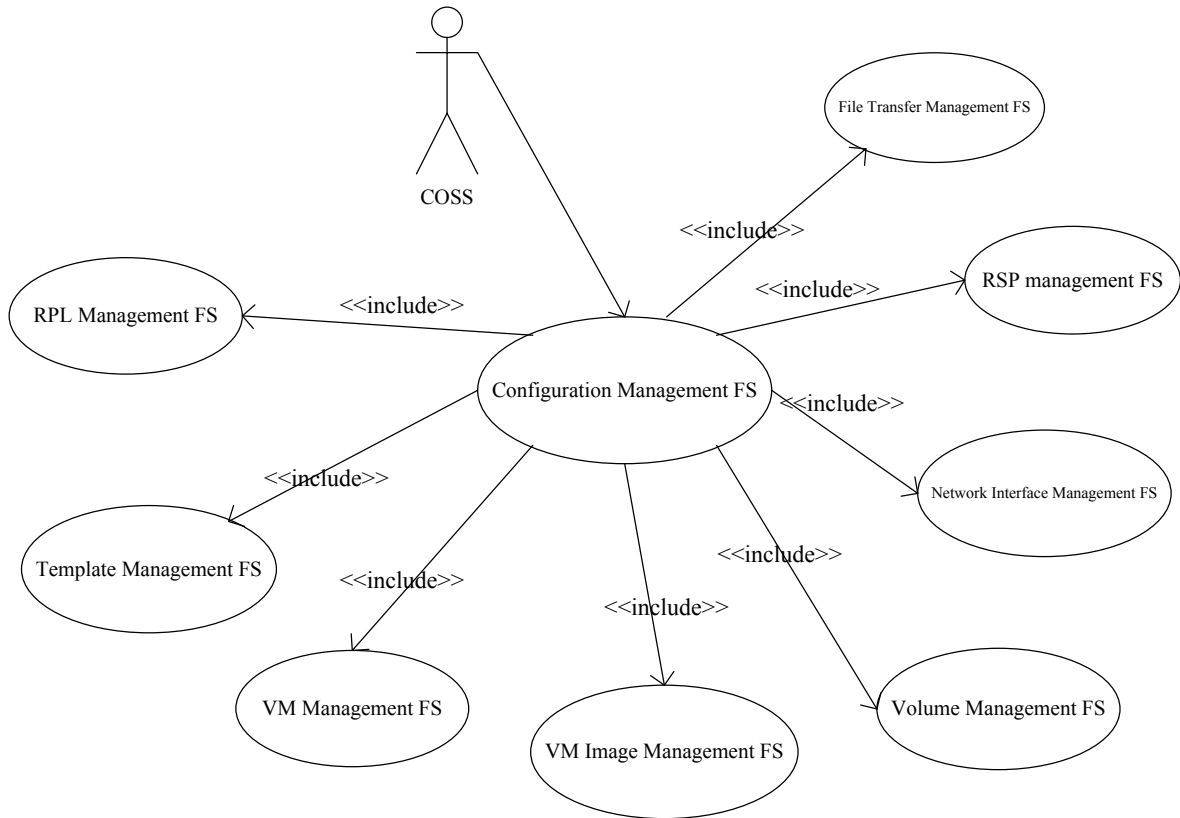
The first overview use case diagram (Figure 7-1) shows the management interface UML use case diagrams.



**Figure7-1 – High-level use case diagram of the management interface**

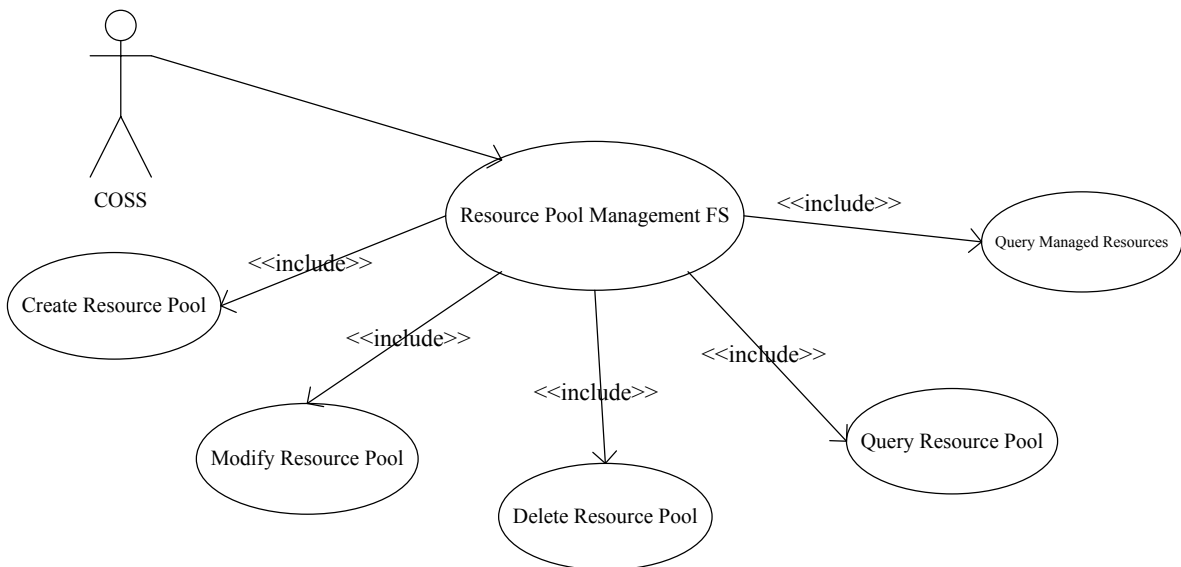
Figure 7-2 shows the configuration management function sets (FS) involved in the management interface.





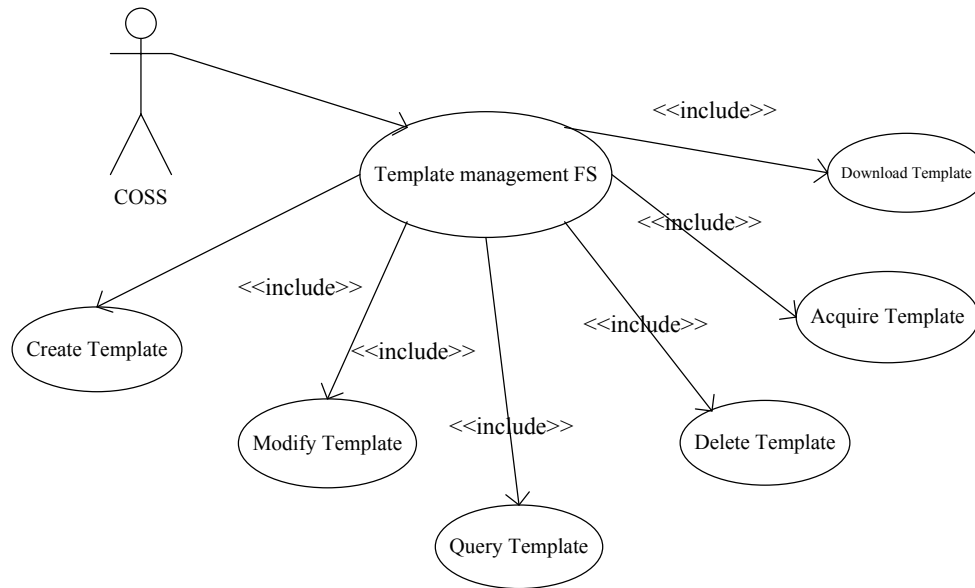
**Figure7-2 – High-level use case diagram of the configuration management interface**

Figure 7-3 shows the Resource Pool management function sets (FS) involved in the management interface.



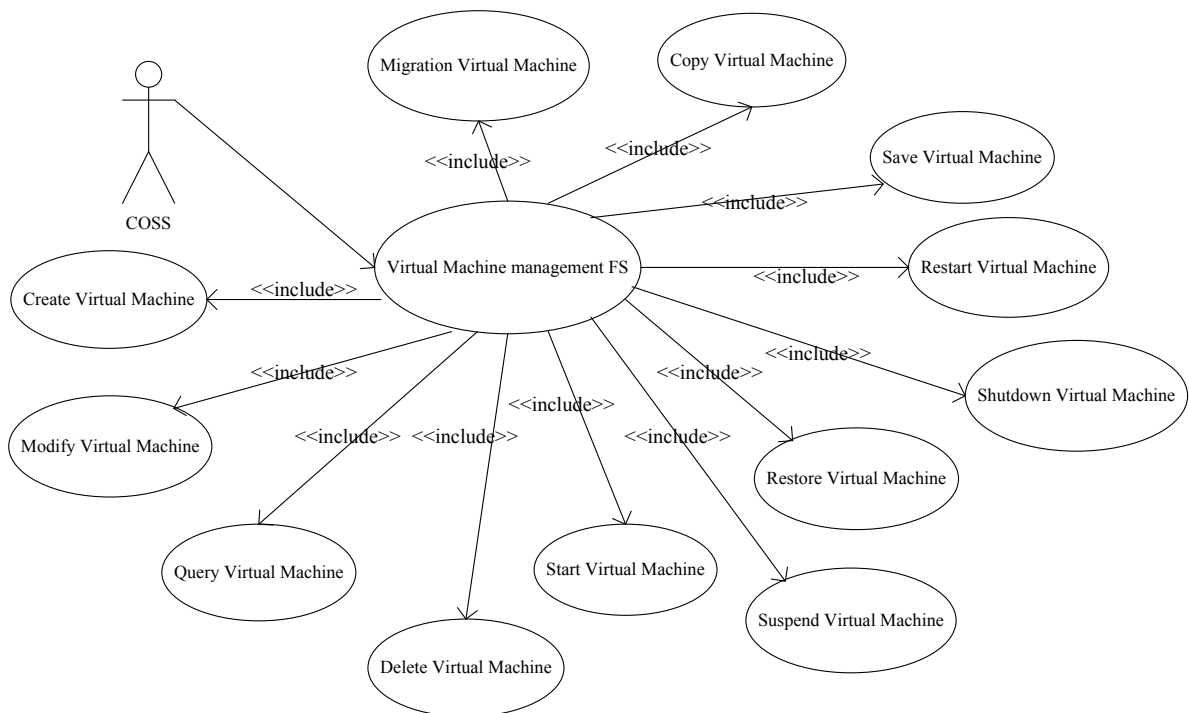
**Figure7-3 – High-level use case diagram of the Resource Pool management interface**

Figure 7-4 shows the template management function sets (FS) involved in the management interface.



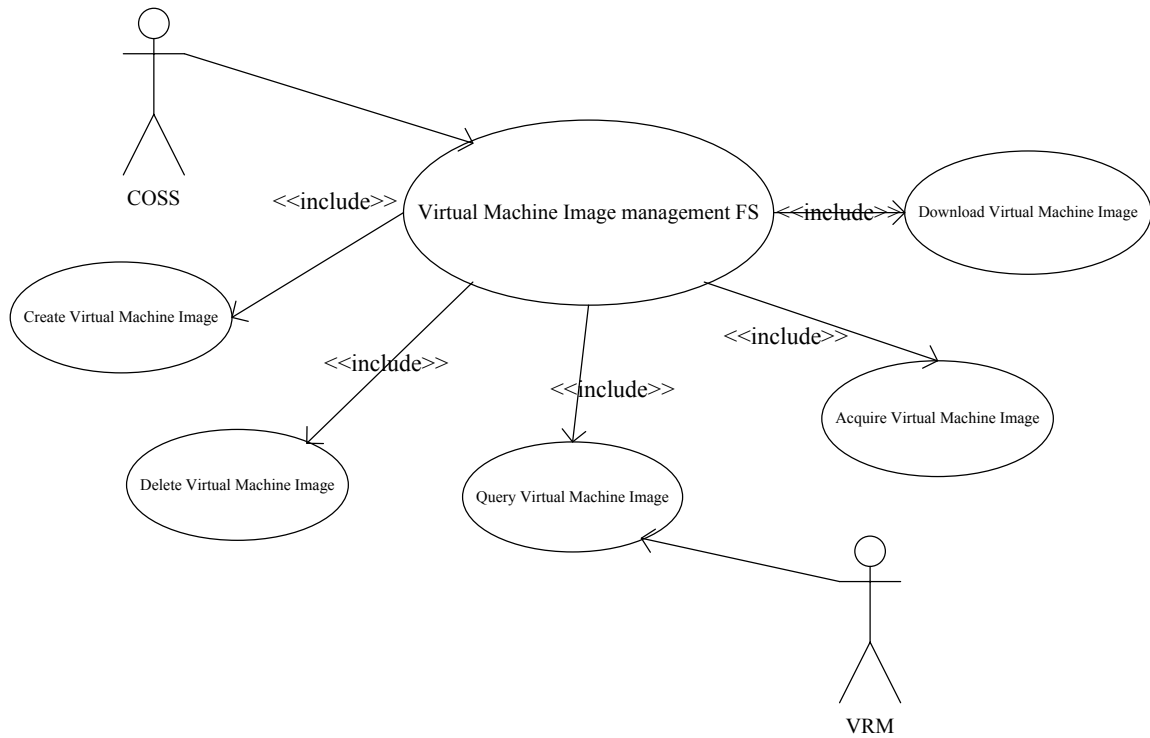
**Figure7-4 – High-level use case diagram of the Template management interface**

Figure 7-5 shows the VM management function sets (FS) involved in the management interface.



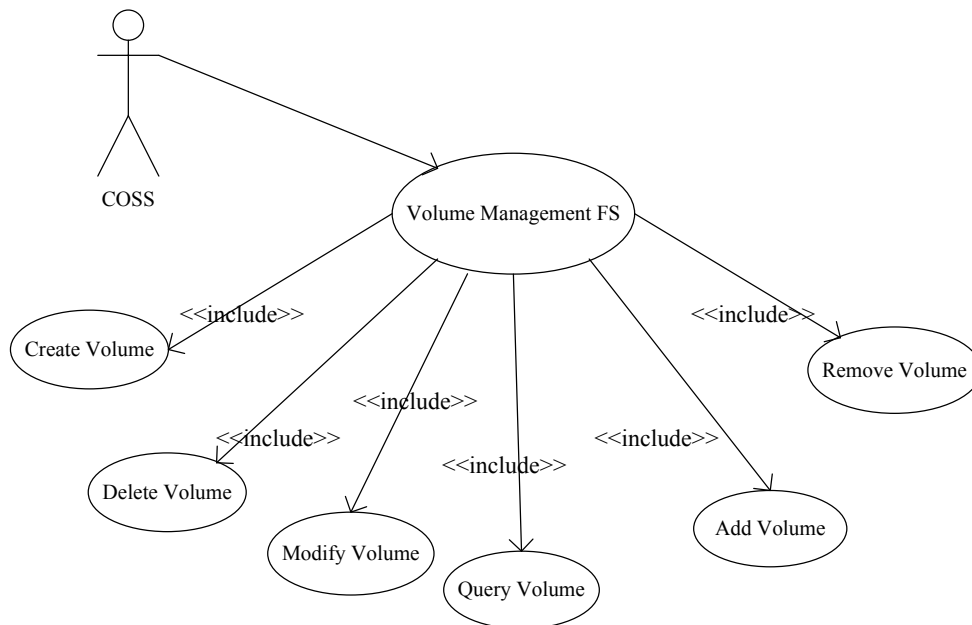
**Figure7-5 – High-level use case diagram of the Virtual Machine management interface**

Figure 7-6 shows the VM Image management function sets (FS) involved in the management interface.



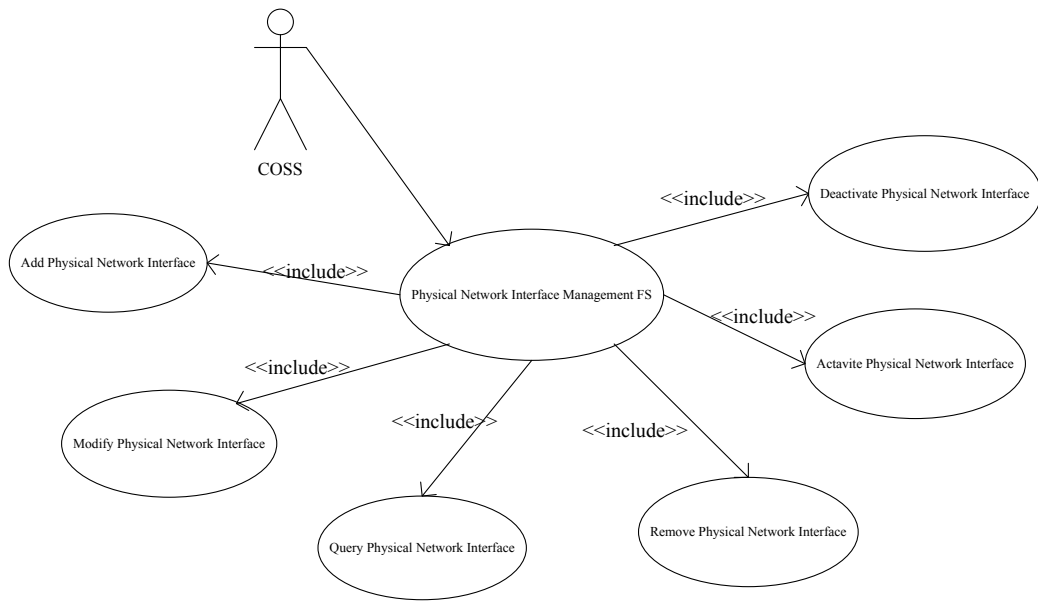
**Figure7-6 – High-level use case diagram of the virtual machine image management interface**

Figure 7-7 shows the Volume management function sets (FS) involved in the management interface.



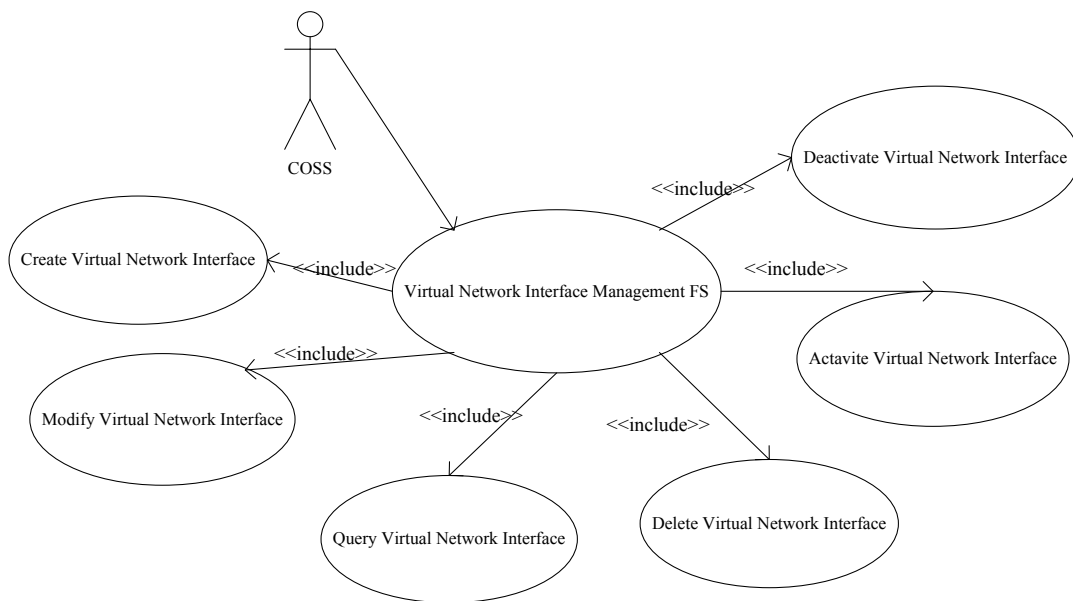
**Figure7-7 – High-level use case diagram of the volume management interface**

Figure 7-8 shows the physical network interface management function sets (FS) involved in the management interface.



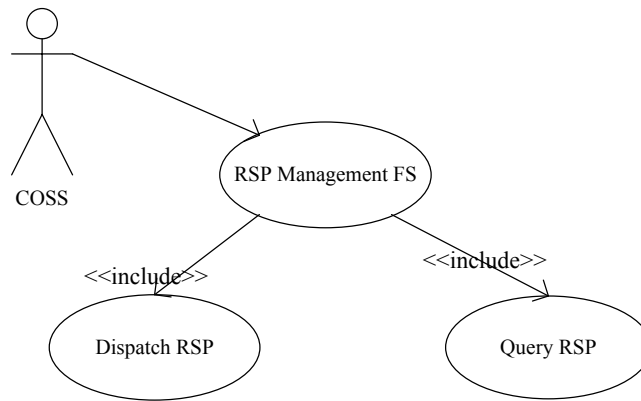
**Figure7-8 – High-level use case diagram of the physical network interface management interface**

Figure 7-9 shows the virtual network interface management function sets (FS) involved in the management interface.



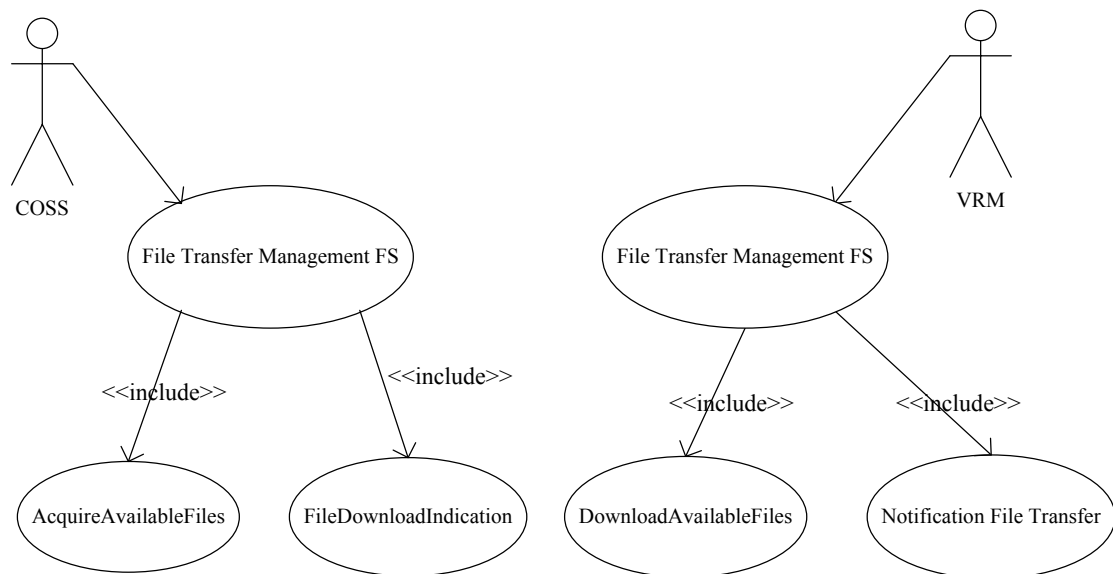
**Figure7-9 – High-level use case diagram of the virtual network interface management interface**

Figure 7-10 shows the Resource scheduling policy management function sets (FS) involved in the management interface.



**Figure7-10 – High-level use case diagram of resource scheduling policy management interface**

Figure 7-11 shows the File transfer management function sets (FS) involved in the management interface.



**Figure7-11 – High-level use case diagram of File transfer management interface**

## 7.2 Specification-level requirements

### 7.2.1 Requirements

There are no specification-level requirements.

### 7.2.2 Actor roles

See clause 7.1.2.

### 7.2.3 Telecommunications resources

See clause 7.1.3.

## **7.2.4 Use cases**

The general exceptions (e.g., communication error, processing error) related to all use cases will not be described in the following uses cases, and they will only be handled in design phases.

### **7.2.4.1 Configuration Management**

The use case related to basic object management including Create object, Modify object, Delete object and Retrieve object information in this Recommendation will reuse the use cases defined in [ITU-T Rec. M.3700]. Only parameters needed for creation, modification and query of different MOI will be specified in this Recommendation.

#### **7.2.4.1.1 Resource Pool Management**

The information needed for creation of a RPL object include template ID or resource pool Id, description, maximum capacity, minimum capacity.

The information of RPL which could be modified include resource pool ID and description.

The information could be retrieved about RPL include resource pool Id, sub-resource pool, description, maximum capacity, minimum capacity, creating time and the RPL status.

#### **7.2.4.1.2 Template Management**

##### **7.2.4.1.2.1 Basic object use cases**

Template management involves the management of RPL template, VM template and Volume template.

The information needed for creation of a RPL template object include maximum capacity, minimum capacity and so on. The information of RPL template which could be modified include maximum capacity, minimum capacity and so on. The information could be retrieved about RPL template include template ID, maximum capacity, minimum capacity, creating time and RPL status.

The information needed for creation of a VM template object include the configuration of a VM, such as OS, CPU, memory, disks and so on which will be used by the VM. The information of VM template which could be modified includes OS, CPU, memory, disks and so on. The information could be retrieved about VM template include template ID, OS, CPU, memory, disks, creating time and VM status.

The information needed for creation of a Volume template object include size and so on.

The information of Volume template which could be modified includes size and so on. The information could be retrieved about Volume template include template ID, size, creating time and volume status.

##### **7.2.4.1.2.2 Acquire Template**

The COSS could acquire the templates stored in the VRM in file format using some file transfer mechanisms. This usecase reuses usecases defined in the clause 7.2.4.1.9.1 in File transfer management.

#### 7.2.4.1.2.2 Download Template

The VRM could download the templates stored in the COSS in file format using some file transfer mechanisms. This usecase reuses **usecases defined** in the clause 7.2.4.1.9.2 **in** File transfer management.

#### 7.2.4.1.3 Virtual Machine Management

##### 7.2.4.1.3.1 Basic object use cases

The information needed for creation of a Virtual Machine object include resource pool ID, OS, CPU, memory, disks and physical network Interface Id and so on.

The information of Virtual Machine which could be modified include CPU, memory, disks and physical network Interface Id or virtual network interface, **priority**, and so on.

The information could be retrieved about Virtual Machine include vmId, resource pool ID, OS, CPU, memory, disks, physical network Interface Id or virtual network interface, priority, creating time and the VM status.

##### 7.2.4.1.3.2 Start Virtual Machine

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to start a specified VM through the management interface.	
<b>Actor and roles</b>	All the VRM or VRs and COSS.	
Telecom resources	VM.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the specified VM is at created status.	
Begins when	The COSS sends a request to a VRM or VR in order to start the VM.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID should be specified in the request.	
Step 2	The specified VM is then started.	
Step 3	The VM is started successfully, the status of the VM is changed to “Started”, and the COSS will receive the operation result as a response. Otherwise, exception is returned to COSS.	
Ends when	The VM is started successfully and the COSS receives the response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) Forbidden	

Use case stage	Evolution/Specification	<<Uses>> Related use
Post-conditions	The VM is started successfully, the status of the VM is changed to “Started”, and the COSS receives the response. Otherwise, exception is returned to COSS..	
Traceability	REQ-CC-FUN-16	

#### 7.2.4.1.3.3 Suspend Virtual Machine

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to suspend a specified VM through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	VM.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the specified VM is at “started” status.	
Begins when	The COSS sends a request to a VRM or VR in order to suspend the VM.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID should be specified in the request.	
Step 2	The specified VM is then suspended.	
Step 3	The VM is suspended successfully, the status of the VM is changed to “Suspended”, and the COSS will receive the operation result as a response. Otherwise, exception is returned to COSS.	
Ends when	The VM is suspended successfully, and the COSS receives the response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) Forbidden	
Post-conditions	The VM is suspended successfully, the status of the VM is changed to “Suspended”, and the COSS receives the response. Otherwise, exception is returned to COSS.	
Traceability	REQ-CC-FUN-17	

#### 7.2.4.1.3.4 Restore Virtual Machine

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to restore a specified VM through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	



Use case stage	Evolution/Specification	<<Uses>> Related use
Telecom resources	VM.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the status of the specified VM is at “Suspended”.	
Begins when	The COSS sends a request to a VRM or VR in order to restore the VM.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID should be specified in the request.	
Step 2	The specified VM is then restored.	
Step 3	The VM is restored successfully, the status of the VM is changed to “Started”, and the COSS will receive the operation result as a response. Otherwise, exception is returned to COSS.	
Ends when	The VM is restored successfully, and the COSS receives the response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) Forbidden	
Post-conditions	The VM is restored successfully, and the VM is at “Started” status. The COSS receives the response. Otherwise, exception is returned to COSS.	
Traceability	REQ-CC-FUN-18	

#### 7.2.4.1.3.5 Shutdown Virtual Machine

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to shut down a specified VM through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	VM.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the status of the VM is at “Started”.	
Begins when	The COSS sends a request to a VRM or VR in order to shut down the VM.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID should be specified in the request.	
Step 2	The specified VM is then shutdown.	
Step 3	The VM is shutdown successfully, the status of the VM is changed to “Stopped”, and the COSS will	

Use case stage	Evolution/Specification	<<Uses>> Related use
	receive the operation result as a response. Otherwise, exception is returned to COSS.	
Ends when	The VM is shutdown successfully, and the COSS receives the response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) Forbidden	
Post-conditions	The VM is shutdown successfully, the VM is at “Stopped” status, and the COSS receives the response. Otherwise, exception is returned to COSS.	
Traceability	REQ-CC-FUN-19	

#### 7.2.4.1.3.6 Restart Virtual Machine

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to restart a specified VM through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	VM.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication and the VM is in the “Stopped” status.	
Begins when	The COSS sends a request to a VRM or VR in order to restart the VM.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID should be specified in the request.	
Step 2	The specified VM is then restarted.	
Step 3	The VM is restarted successfully, the Status of the VM is changed to “Started”, and the COSS will receive the operation result as a response. Otherwise, exception is returned to COSS.	
Ends when	The VM is restarted successfully, and the COSS receives the response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) Forbidden	
Post-conditions	The VM is restarted successfully, and the VM is at “Started” status. The COSS receives the response. Otherwise, exception is returned to COSS..	
Traceability	REQ-CC-FUN-20	

#### 7.2.4.1.3.7 Save Virtual Machine

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to save a specified VM and the snapshot information would be returned through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	VM.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the VM is at “Suspended” status.	
Begins when	The COSS sends a request to a VRM or VR in order to save the specified VM.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID should be specified in the request.	
Step 2	The specified VM is then saved, and the result is returned to the COSS as the response.	
Step 3	The VM is saved successfully, and the COSS will receive the operation result as a response. Otherwise, exception is returned to COSS.	
Ends when	The VM is saved successfully, and the COSS receives the snapshot information as a response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) Forbidden	
Post-conditions	The VM is saved successfully, and the COSS receives the response. Otherwise, exception is returned to COSS.	
Traceability	REQ-CC-FUN-21	

#### 7.2.4.1.3.8 Copy Virtual Machine

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	Copy VM is also called clone VM. The COSS can send a request to the VRM or VR to copy a specified VM and get another VM through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	VM.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the specified VM, as well as the physical server which is used to store the new VM should exist.	

Use case stage	Evolution/Specification	<<Uses>> Related use
Begins when	The COSS sends a request to a VRM or VR in order to copy the specified VM.	
Step 1	The COSS sends a request to a VRM or VR, and VM ID, New VM ID, Physical server ID should be specified in the request.	
Step 2	The specified VM is then copied and the new VM is stored to the specified physical server.	
Step 3	The VM is copied successfully, and the new VM is stored to the specified physical server, the COSS will receive the operation result as a response. Otherwise, exception is returned to COSS.	
Ends when	The VM is copied successfully, and the new VM is stored to the specified physical server. The COSS receives the response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) Forbidden	
Post-conditions	The VM is copied successfully, and the new VM is stored to the specified physical server. The COSS receives the response. Otherwise, exception is returned to COSS.	
Traceability	REQ-CC-FUN-22	

#### 7.2.4.1.3.9 Migrate Virtual Machine

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The VRM or VR should be able to migrate a specified VM from one physical server to another physical server through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	VM.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the specified VM, as well as the physical server which is used to store the migrated VM should exist.	
Begins when	The COSS sends a request to a VRM or VR in order to migrate the VM.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID, Source Location, Destination Location should be specified in the request.	
Step 2	The specified VM is then migrated to the destination physical server.	

Use case stage	Evolution/Specification	<<Uses>> Related use
Step 3	The VM is migrated to the destination physical server successfully, and the COSS will receive the operation result as a response. Otherwise, exception is returned to COSS, and the VM rollbacks to the initial state.	
Ends when	The VM is migrated successfully, and the COSS receives the response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) Forbidden	
Post-conditions	The VM is migrated to the destination physical server successfully, and the COSS receives the response. Otherwise, exception is returned to COSS.	
Traceability	REQ-CC-FUN-23	

#### 7.2.4.1.4 Virtual Machine Image Management

##### 7.2.4.1.4.1 Basic object management use cases

The information needed for creation of a Virtual Machine Image object include the vmID<sub>[WY11]</sub> and the image name which will be generated.

The information could be retrieved about Virtual Machine Image include image ID, image name, os and creating time.

##### 7.2.4.1.4.2 Acquire Virtual Machine Image

The COSS could acquire the virtual machine images stored in the VRM in file format using some file transfer mechanisms. This usecase reuses usecases defined in the clause 7.2.4.1.9.1 in File transfer management.

##### 7.2.4.1.2.2 Download Virtual Machine Image

The VRM could download the virtual machine images stored in the COSS in file format using some file transfer mechanisms. This usecase reuses usecases defined in the clause 7.2.4.1.9.2 in File transfer management.

#### 7.2.4.1.5 Volume Management

##### 7.2.4.1.5.1 Basic object management use cases

The information needed for creation of a Volume object include resourcePoolID, volume size and Location.

The information of Volume which could be modified includes size and description.

The information could be retrieved about Volume include volumeId, resourcePoolID, volume size and Location, creating time and the volume status.

#### 7.2.4.1.5.2 Attach Volume

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to attach the specified volume to a specified VM through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	Volume.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the specified VM should exist.	
Begins when	The COSS sends a request to a VRM or VR in order to add the volume to the specified VM.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID, volume ID .etc. should be specified in the request.	
Step 2	The volume is then attached to the VM.	
Step 3	The volume is attached to the specified VM successfully, and the COSS will receive the response. Otherwise, exception is returned to COSS.	
Ends when	The volume is attached to the specified VM successfully, and the COSS receives the response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) No Content	
Post-conditions	The volume is attached to the specified VM successfully, and the COSS receives the response. Otherwise, exception is returned to COSS.	
Traceability	REQ-CC-FUN-33	

#### 7.2.4.1.5.3 Detach Volume

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to detach the specified volume from a specified VM through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	Volume.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and	

Use case stage	Evolution/Specification	<<Uses>> Related use
	the specified VM should exist.	
Begins when	The COSS sends a request to a VRM or VR in order to detach the volume from the VM.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID, volume ID .etc. should be specified in the request.	
Step 2	The volume is then detached from the VM.	
Step 3	The specified volume is detached from the VM successfully, and the COSS will receive the response. Otherwise, exception is returned to COSS.	
Ends when	The specified volume is detached successfully, and the COSS receives the response.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) No Content	
Post-conditions	The specified volume is detached successfully, and the COSS receives the response. Otherwise, exception is returned to COSS.	
Traceability	REQ-CC-FUN-34	

#### 7.2.4.1.6 Physical Network Interface Management

##### 7.2.4.1.6.1 Basic object management use cases

The information needed for creation of a physical network interface object include IP address and MAC address.

The information of physical network interface which could be modified include physical network interface ID and IP Address.

The information could be retrieved about physical network interface include physical network interface ID, IP Address, creating time and the physical network interface status.

##### 7.2.4.1.6.2 Activate Network Interface

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to activate a physical network interface through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	Physical network interface.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the specified VM and physical network interface should exist. Moreover, the status of the physical	

Use case stage	Evolution/Specification	<<Uses>> Related use
	network interface is at “Created” or “Inactivated”.	
Begins when	The COSS sends a request to a VRM or VR in order to activate a physical network interface.	
Step 1	The COSS sends a request to a VRM or VR, and the physical network interface ID should be specified in the request.	
Step 2	The physical network interface is then activated.	
Step 3	The specified physical network interface is activated successfully, the physical network interface should be at “activated” status, and the result will be returned to the COSS as a response. Otherwise, exception is returned to COSS.	
Ends when	The specified physical network interface is started successfully, and the COSS receives the operation response.	
Exceptions	1) Not Implemented 2) Not Modified 3) Not Found 4) Method Not Allowed 5) Conflict	
Post-conditions	The specified physical network interface is started successfully, and the physical network interface is at “activated” status. The COSS receives the operation response. Otherwise, the exception is returned to COSS.	
Traceability	REQ-CC-FUN-39	

#### 7.2.4.1.6.3 Deactivate Network Interface

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to deactivate the physical network interface through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	Physical network interface.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication. The specified VM and physical network interface should exist, and the physical network interface is at “activated” status.	
Begins when	The COSS sends a request to a VRM or VR in order to deactivate a physical network interface.	
Step 1	The COSS sends a request to a VRM or VR, and the physical network interface ID should be specified in the request.	



Use case stage	Evolution/Specification	<<Uses>> Related use
Step 2	The network interface is then deactivated.	
Step 3	The specified physical network interface is deactivated successfully, and the physical network interface should be at “inactivated” status. The result will be returned to the COSS as a response. Otherwise, exception is returned to COSS.	
Ends when	The specified physical network interface is deactivated, and the COSS receives the operation response.	
Exceptions	1) Not Implemented 2) Not Modified 3) Not Found 4) Method Not Allowed 5) Conflict	
Post-conditions	The specified physical network interface is deactivated successfully, and the physical network interface is at “inactivated” status. The COSS receives the operation response. Otherwise, the exception is returned to COSS.	
Traceability	REQ-CC-FUN-40	

#### 7.2.4.1.7 Virtual Network Interface Management

##### 7.2.4.1.7.1 Basic object management use cases

The information needed for creation of a virtual network interface object include physical network interface ID, VM ID, MAC address and maxLimitRate.

The information of virtual network interface which could be modified include physical network interface ID, MAC address and maxLimitRate.

The information could be retrieved about virtual network interface include virtual network interface ID, physical network interface ID, IP Address, MAC address, maxLimitRate, creating time and the virtual network interface status.

##### 7.2.4.1.7.2 Activate Network Interface

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to activate a virtual network interface through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	Virtual network interface.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the specified VM and virtual network interface	

Use case stage	Evolution/Specification	<<Uses>> Related use
	should exist. Moreover, the virtual network interface is at “Created” or “Inactivated” status.	
Begins when	The COSS sends a request to a VRM or VR in order to activate a virtual network interface.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID, virtual network interface ID should be specified in the request.	
Step 2	The virtual network interface is then activated.	
Step 3	The specified virtual network interface is activated successfully, the virtual network interface status should be “activated”, and the result will be returned to the COSS as a response. Otherwise, exception is returned to COSS.	
Ends when	The specified virtual network interface is activated successfully, and the COSS receives the operation response.	
Exceptions	1) Not Implemented 2) Not Modified 3) Not Found 4) Method Not Allowed 5) Conflict	
Post-conditions	The specified virtual network interface is activated successfully, and the status of the virtual network interface is at “activated”. The COSS receives the operation response. Otherwise, the exception is returned to COSS.	
Traceability	REQ-CC-FUN-45	

#### 7.2.4.1.7.3 Deactivate Network Interface

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM or VR to deactivate the virtual network interface through the management interface.	
Actor and roles	All the VRM or VRs and COSS.	
Telecom resources	Virtual network interface.	
Assumptions	The communication between the VRM or VR and COSS is available.	
Pre-conditions	The COSS has built an interface communication. The specified VM and virtual network interface should exist, and the status of the virtual network interface is at “activated”.	
Begins when	The COSS sends a request to a VRM or VR in order to deactivate a virtual network interface.	
Step 1	The COSS sends a request to a VRM or VR, and the VM ID, virtual network interface ID should be	

Use case stage	Evolution/Specification	<<Uses>> Related use
	specified in the request.	
Step 2	The virtual network interface is then deactivated.	
Step 3	The specified virtual network interface is deactivated successfully, the status of the virtual network interface should be at “inactivated”. The result will be returned to the COSS as a response. Otherwise, exception is returned to COSS.	
Ends when	The specified virtual network interface is deactivated, and the COSS receives the operation response.	
Exceptions	1) Not Implemented 2) Not Modified 3) Not Found 4) Method Not Allowed 5) Conflict	
Post-conditions	The specified virtual network interface is deactivated successfully, and the status of the virtual network interface is at “inactivated” status. The COSS receives the operation response. Otherwise, the exception is returned to COSS.	
Traceability	REQ-CC-FUN-46	

#### 7.2.4.1.8 Resource Scheduling Policy Management

##### 7.2.4.1.8.1 Basic object management use cases [WY12]

The information could be retrieved about Resource Scheduling Policy include resource policy ID and resource policy description.

##### 7.2.4.1.8.2 Dispatch Resource Scheduling Policy

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM to dispatch the resource scheduling policy through the management interface.	
Actor and roles	All the VRM and COSS.	
Telecom resources	Resource scheduling policy.	
Assumptions	The communication between the VRM and COSS is available.	
Pre-conditions	The COSS has built an interface communication.	
Begins when	The COSS sends a request to a VRM in order to dispatch the resource scheduling policy.	
Step 1	The COSS sends a request to a VRM to dispatch the resource scheduling policy. The following parameters should be specified in the request:	

Use case stage	Evolution/Specification	<<Uses>> Related use
	- Resource scheduling policy information (such as resourcePolicyID, resourcePolicyDescription),etc.	
Step 2	The resource scheduling policy is then dispatched to the VRM.	
Step 3	If The resource scheduling policy is dispatched to the VRM successfully, and the result will be returned to the COSS as a response. Otherwise, exception is returned to COSS.	
Ends when	The resource scheduling policy is dispatched to the VRM successfully, and the COSS receives the operation response.	
Exceptions	1) Not Implemented 2) Forbidden 3) Method Not Allowed	
Post-conditions	The resource scheduling policy is dispatched to the VRM successfully, and the COSS receives the operation response. Otherwise, exception is returned to COSS.	
Traceability	REQ-CC-FUN-48	

#### 7.2.4.1.9 File Transfer Management

##### 7.2.4.1.9.1 AcquireAvailableFiles

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to the VRM to acquire the available file(s) through the management interface. On receiving this request, the VRM will prepare the file(s). When the files are ready, the COSS will be notified and will then retrieve those files using some file transfer mechanisms, e.g., FTP.	
Actor and roles	All the VRM and COSS.	
Telecom resources	The template file(s) or virtual machine image file(s).	
Assumptions	The communication between the VRM and COSS is available.	
Pre-conditions	The COSS has built an interface communication.	
Begins when	The COSS sends a request to VRM in order to acquire the file(s).	
Step 1	The COSS sends a request to a VRM, and the input parameters are as following: – object type, includes template file(s) or image file(s), – object Id list, etc.	
Step 2	A response which indicates whether the operation is	

Use case stage	Evolution/Specification	<<Uses>> Related use
	successful will be returned to the COSS.	
Ends when	A response or an exception is returned to the COSS.	
Exceptions	1) Not Implemented 2) Not Found 3) Method Not Allowed 4) No Content	
Post-conditions	The VRM starts a file preparation task on the request of the COSS. When the preparation is finished, the VRM will send a "NotifyFileTransferReady" notification to the COSS. When the preparation is failed, the VRM will send a "NotifyFilePreparationError" notification to the COSS.	
Traceability	REQ-CC-FUN-49	

#### 7.2.4.1.9.2 DownloadAvailableFiles

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The VRM can send a request to the COSS to download file(s) through the management interface. On receiving this request, the COSS will prepare the file(s). When the files are ready, the COSS will invoke the operation "FileDownloadIndication" to indicate the VRM to download the file(s) and then the VRM will retrieve those files using file transfer mechanisms, e.g., FTP.	
Actor and roles	All the VRM and COSS.	
Telecom resources	The template file(s) or virtual machine image file(s).	
Assumptions	The communication between the VRM and COSS is available.	
Pre-conditions	The COSS has built an interface communication.	
Begins when	The VRM sends a request to the COSS in order to download the file(s).	
Step 1	The VRM sends a request to the COSS, and the input parameters are as following: – object type, includes template file(s) or image file(s), – object Id list, etc.	
Step 2	A response which indicates whether the operation is successful will be returned to the VRM.	
Ends when	A response or an exception is returned to the VRM.	
Exceptions	1) Not Implemented	

Use case stage	Evolution/Specification	<<Uses>> Related use
	2) Not Found 3) Method Not Allowed 4) No Content	
Post-conditions	The COSS starts a file preparation task on the request of the VRM. When the preparation is finished, the COSS will invoke "FileDownloadIndication" to indicate the VRM to download the files.	
Traceability	REQ-CC-FUN-50	

#### 7.2.4.1.9.3 FileDownloadIndication

Use case stage	Evolution/Specification	<<Uses>> Related use
Goal	The COSS can send a request to indicate the VRM to download files through the management interface.	
Actor and roles	All the VRM and COSS.	
Telecom resources	Template files or virtual machine image files.	
Assumptions	The communication between the VRM and COSS is available.	
Pre-conditions	The COSS has built an interface communication, and the fileInforList exists.	
Begins when	The COSS sends a request to the VRM in order to indicate the VRM to download the files.	
Step 1	The COSS sends a request to the VRM, and the input parameters contain: – fileDirectory, – the sequence of fileInforList. The fileInforList is defined as following: – fileName, – fileSize, – fileCompression, – fileCreationTime, – fileDeletionTime.	
Step 2	The file download indication information is sent to VRM successfully, and the COSS receives the operation response.	
Ends when	The file download indication information is sent to VRM successfully.	
Exceptions	1) Not Implemented 2) Not Found 3) No Content	
Post-conditions	The file is downloaded successfully by the VRM according to the information from "FileDownloadIndication".	

Use case stage	Evolution/Specification	<<Uses>> Related use
Traceability	REQ-CC-FUN-50	

Editor's notes: how to treat notifications?

#### 7.2.4.2 Fault Management

The use cases of the fault management reuse the definition in the [ITU-T Rec M.3703].

~~For the probable cause of the virtual resource in the cloud computing environment, please to 7.1.1.2.~~

#### 7.2.4.3 Performance Management

The use cases of the performance management reuse the definition in the [ITU-T Rec M.3704].

For the performance parameters of the VM, volume and network interface, please refer to 7.1.1.3.

## 8 Analysis

### 8.1 Information object classes

#### 8.1.1 Information entities imported and local label

Label reference	Local label
[ITU-T M.3160], information object class, Top	Top

#### 8.1.2 Class diagram

##### 8.1.2.1 Attributes and relationships

Figure 8-1 shows the class containment relationships of information object classes defined in the Recommendation.

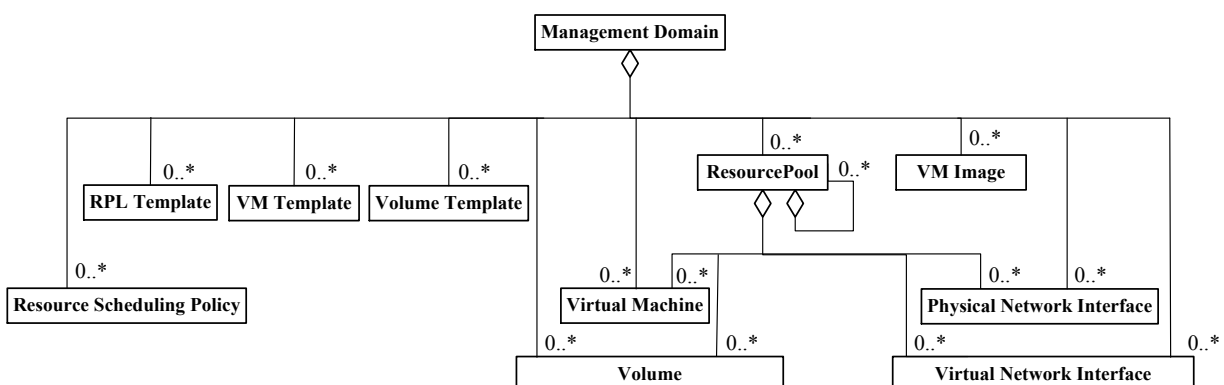
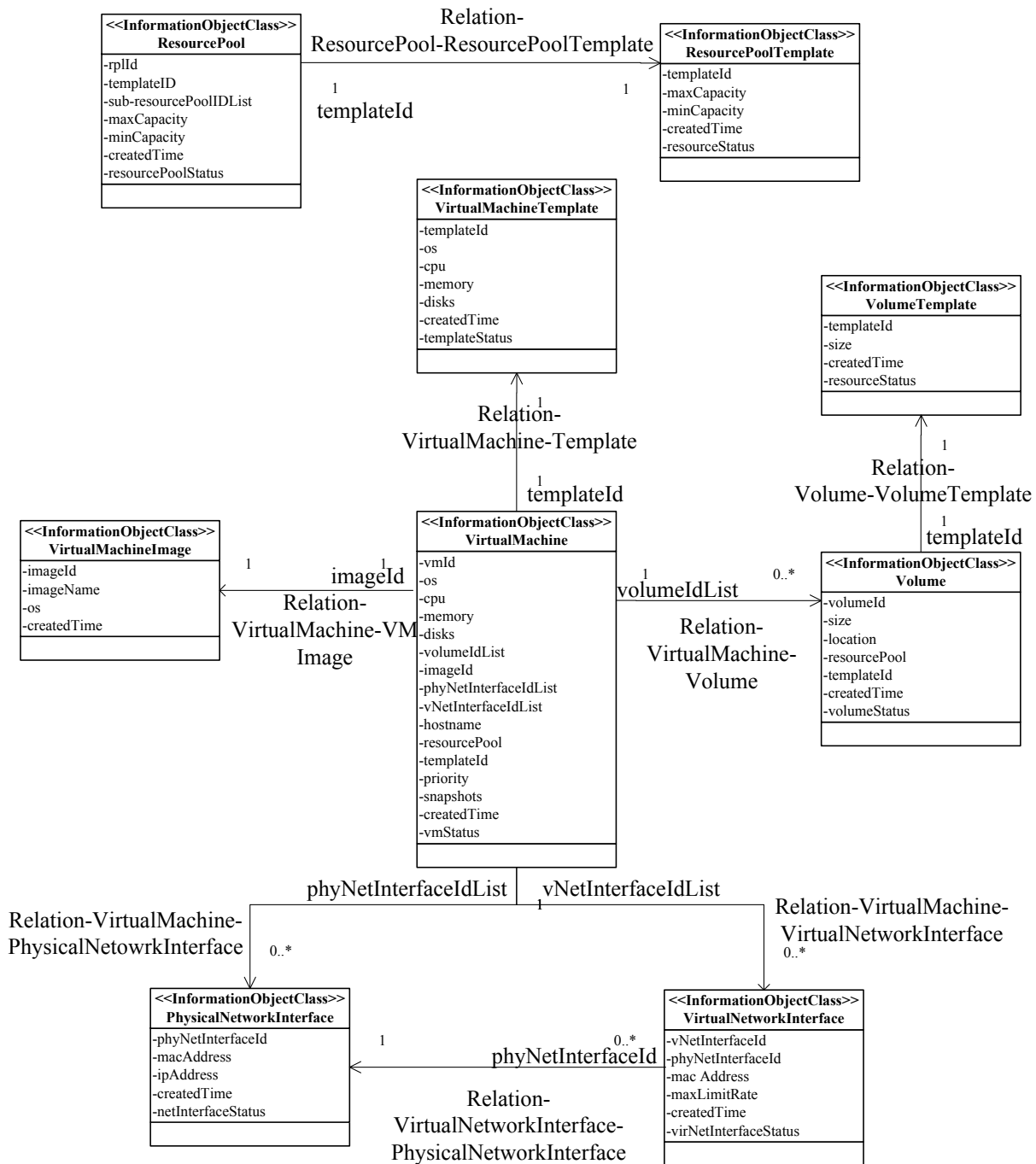


Figure 8-1 Containment diagram of IOCs

Figure 8-2 shows the detailed ER diagram of **instances of** information object classes VM, Template, virtual Machine Image, Volume, Physical Network Interface, and Virtual Network Interface.



**Figure 8-2 – ER diagram**

### 8.1.2.2 Inheritance

Figure 8-3 shows the inheritance diagram of IOCs defined in this **draft** Recommendation.



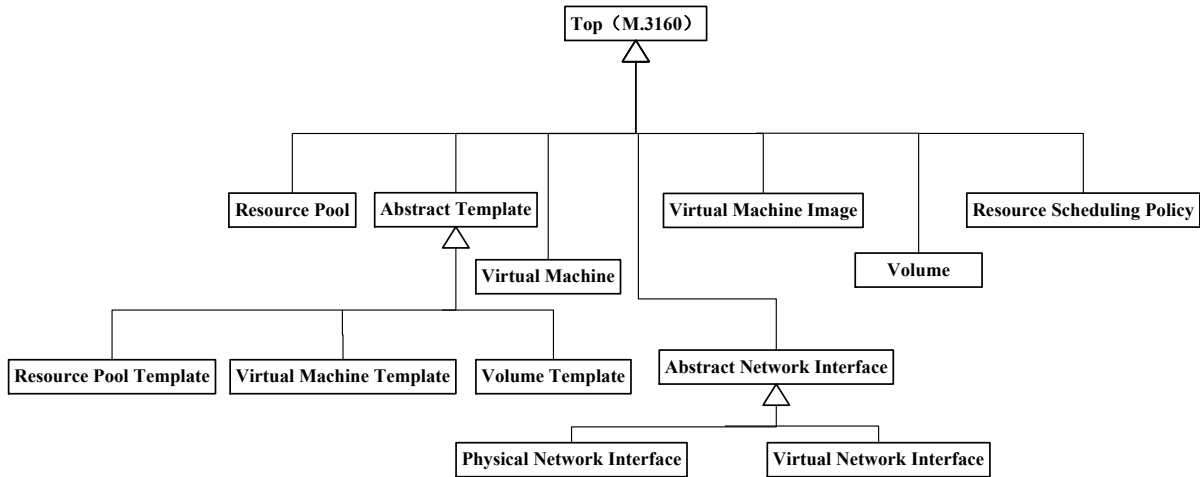


Figure 8-3 – Inheritance diagram

### 8.1.3 Information object class definitions

Class name	Qualifier	Requirement IDs
Resource Pool	M	REQ-CC-FUN-01,REQ-CC-FUN-02 REQ-CC-FUN-03,REQ-CC-FUN-04 REQ-CC-FUN-05
Template	M	REQ-CC-FUN-01, , REQ-CC-FUN-06,REQ-CC-FUN-07 REQ-CC-FUN-08,REQ-CC-FUN-09 REQ-CC-FUN-10, REQ-CC-FUN-11 REQ-CC-FUN-12, REQ-CC-FUN-29 REQ-CC-FUN-49, REQ-CC-FUN-50
Virtual Machine	M	REQ-CC-FUN-12,REQ-CC-FUN-13, REQ-CC-FUN-14, REQ-CC-FUN-15, REQ-CC-FUN-16 REQ-CC-FUN-17, REQ-CC-FUN-18, REQ-CC-FUN-19, REQ-CC-FUN-20, REQ-CC-FUN-21 REQ-CC-FUN-22, REQ-CC-FUN-23,
VM Image	M	REQ-CC-FUN-24, REQ-CC-FUN-25 REQ-CC-FUN-26, REQ-CC-FUN-27, REQ-CC-FUN-28, REQ-CC-FUN-49 REQ-CC-FUN-50
Volume	M	REQ-CC-FUN-29, REQ-CC-FUN-30 REQ-CC-FUN-31, REQ-CC-FUN-32 REQ-CC-FUN-33, REQ-CC-FUN-34
Physical Network Interface	M	REQ-CC-FUN-35, REQ-CC-FUN-36 REQ-CC-FUN-37, REQ-CC-FUN-38 REQ-CC-FUN-39, REQ-CC-FUN-40

Virtual Network Interface	M	REQ-CC-FUN-41, REQ-CC-FUN-42 REQ-CC-FUN-43, REQ-CC-FUN-44 REQ-CC-FUN-45, REQ-CC-FUN-46
Resource Scheduling Policy	M	REQ-CC-FUN-47, REQ-CC-FUN-48

### 8.1.3.1 Management Domain

#### 8.1.3.1.1 Definition

The management domain of a cloud operational support system.

#### 8.1.3.1.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
managementDoaminId	M	M	M	
userLable	M	M	M	
managedResourceIdList	O	M	–	

### 8.1.3.2 Resource Pool

#### 8.1.3.2.1 Definition

A resource pool is a set of resources available for assignment to services or applications. Resources in the resource pool can be assigned exclusively or shared by other resource consumer(s).

#### 8.1.3.2.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
rpId	M	M	–	
templateID	O	M	–	
sub-resourcePoolIDList	O	M	–	
maxCapacity	O	M	M	
minCapacity	O	M	M	
createdTime	M	M	–	
resourcePoolStatus	M	M	–	

### 8.1.3.3 Abstract Template

#### 8.1.3.3.1 Definition

Abstract template represents the set of metadata and instructions used to instantiate resources (e.g., a virtual machine template is used to create virtual machines). A template could be a resource pool

template, a VM template or a volume template. Template is an abstract IOC used for inheritance purpose, and should not be instantiated.

#### 8.1.3.3.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
templateId	M	M	—	[WY13]
createdTime	M	M	—	
resourceStatus	M	M	—	

#### 8.1.3.4 Resource Pool Template

##### 8.1.3.4.1 Definition

A resource pool template is a deployable entity to realize a resource pool which could contain multiple resources with their interconnection relationship. This IOC inherits from IOC Template.

##### 8.1.3.4.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
maxCapacity	M	M	M	
minCapacity	M	M	M	

#### 8.1.3.5 Virtual Machine Template

##### 8.1.3.5.1 Definition

A virtual machine template is a preconfigured deployable entity which could realize a VM. This IOC inherits from IOC Template.

##### 8.1.3.5.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
os	M	M	—	
cpu	M	M	M	
memory	M	M	M	
disks	M	M	M	

#### 8.1.3.6 Volume Template

##### 8.1.3.6.1 Definition

A volume template is a preconfigured deployable entity which could realize a volume resource. This IOC inherits from IOC Template.

### 8.1.3.6.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
size WY14	M	M	M	

### 8.1.3.7 Virtual Machine

#### 8.1.3.7.1 Definition

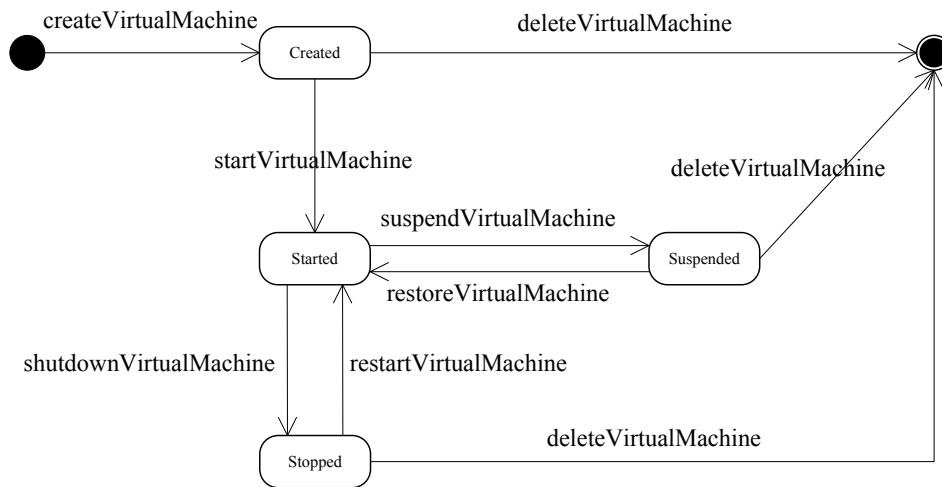
Virtual Machine is the complete environment that supports the execution of guest software. A virtual machine is a full encapsulation of the virtual hardware, virtual disks, and the metadata associated with it. Virtual machines allow multiplexing of the underlying physical machine through a software layer called a hypervisor.

#### 8.1.3.7.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
vmId	M	M	–	
os	MC	M	–	
cpu	M	M	M	
memory	M	M	M	
disks	M	M	M	
volumIdList	M	M	M	
imageId	MC	M	–	
phyNetInterfaceIdList	M	M	M	
vNetInterfaceIdList	O	M	M	
hostname	M	M	–	
resourcePoolId	O	M	–	
templateId	O	M	–	
priority	O	M	M	
snapshots	O	M	–	
createdTime	M	M	–	
vmStatus	M	M	–	

### 8.1.3.7.3 State diagram

Figure 8-5 depicts the states that can be supported by a VM.



**Figure 8-5 – State diagram of VM**

### 8.2.3.8 Virtual Machine Image

#### 8.2.3.8.1 Definition

Virtual Machine Image represents the information necessary for virtual machine to create a Machine Instance.

#### 8.2.3.8.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
imageId	M	M	–	
imageName	M	M	–	
os[WY15]	O	M	–	
createdTime	O	M	–	

### 8.1.3.9 Volume

#### 8.1.3.9.1 Definition

Volume is the storage medium that is associated with a logical disk spanning on one or more hard disk drives. It can be used for interoperable configure storage by the computing cloud.

#### 8.1.3.9.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
volumeId	M	M	–	

size	M	M	M	
location	O	M	–	
resourcePool	M	M	–	
templateId	O	M	–	
createdTime	M	M	–	
volumeStatus	M	M	–	

### 8.1.3.10 Abstract Network Interface

#### 8.1.3.10.1 Definition

Abstract network interface represents the point of interconnection between a computer and a private or public network. Abstract Network Interface is an abstract IOC used for inheritance purpose, and should not be instantiated.

#### 8.1.3.10.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
macAddress	M	M	–	
createdTime	M	M	–	
netInterfaceStatus	M	M	–	

### 8.1.3.11 Physical Network Interface

#### 8.1.3.11.1 Definition

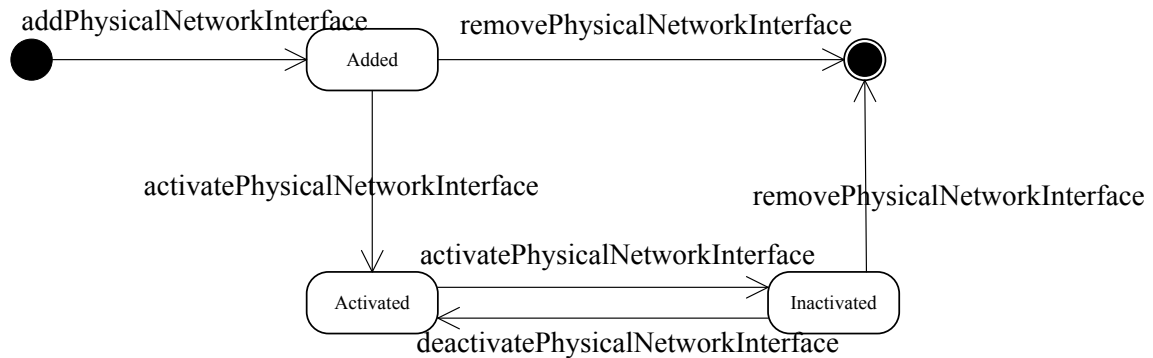
A physical network interface is the physical point of interconnection between a computer and a private or public network. This IOC inherits from IOC Abstract Network Interface.

#### 8.1.3.11.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
phyNetInterfaceId	M	M	M	
ipAddress	M	M	M	

### 8.1.3.9.3 State diagram

Figure 8-6 depicts the states that can be supported by a physical network interface.



**Figure 8-6 – State diagram of physical network interface**

### 8.1.3.12 Virtual Network Interface

#### 8.1.3.12.1 Definition

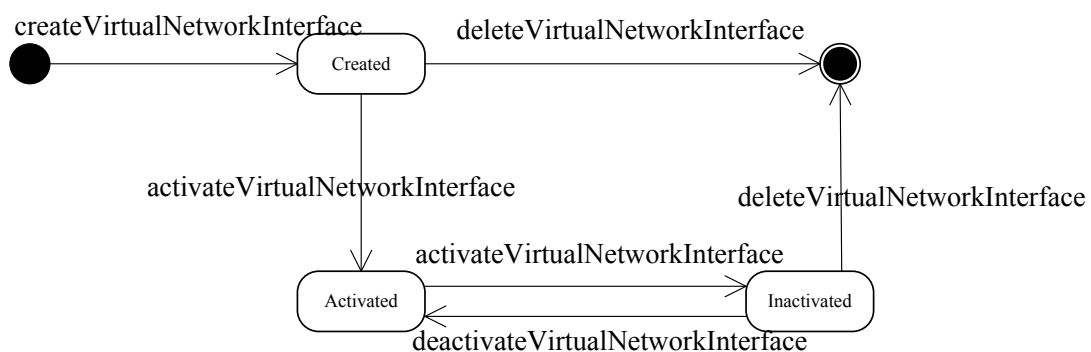
A virtual network interface is the virtual point of interconnection between a computer and a private or public network. This IOC inherits from IOC Abstract Network Interface.

#### 8.1.3.12.2 Attributes

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
vNetInterfaceId	M	M	M	
phyNetInterfaceId	M	M	M	
maxLimitRate	O	M	M	

### 8.1.3.12.3 State diagram

Figure 8-7 depicts the states that can be supported by a virtual network interface.



**Figure 8-7 – State diagram of virtual network interface**

### **8.1.3.13 Resource Scheduling Policy**

#### **8.1.3.13.1 Definition**

Resource Scheduling Policy represents a set of rules that are used to manage and control the state and state transition of one or more virtualized resources.

#### **8.1.3.13.2 Attributes**<sup>[WY16]</sup>

Attribute name	Support qualifier	Read qualifier	Write qualifier	Requirement IDs
resourcePolicyId	M	M	–	
resourcePolicyDescription	M	M	M	

### **8.1.4 Information relationship definitions**

#### **8.1.4.1 Relation ResourcePool- ResourcePoolTemplate (M)**

##### **8.1.4.1.1 Definition**

This represents a unidirectional relation from the ResourcePool to the ResourcePoolTemplate.

The role of the relation should be mapped to a reference attribute of ResourcePool. The name of the reference attribute shall be the role name.

##### **8.1.4.1.2 Roles**

Name	Definition
templateId	This role (when present) represents the template identifier of the ResourcePool which created by using an existing ResourcePool template.

##### **8.1.4.1.3 Constraint**

Name	Definition
–	–

#### **8.1.4.2 Relation VirtualMachine-VirtualMachineTemplate (M)**

##### **8.1.4.2.1 Definition**

This represents a unidirectional relation from the VirtualMachine to the VirtualMachineTemplate.



The role of the relation should be mapped to a reference attribute of VirtualMachine. The name of the reference attribute shall be the role name.

#### 8.1.4.2.2 Roles

Name	Definition
templateId	This role (when present) represents the template identifier of the VM which created by using an existing VM template.

#### 8.1.4.2.3 Constraint

Name	Definition
—	—

### 8.1.4.3 RelationVolume-VolumeTemplate (M)

#### 8.1.4.3.1 Definition

This represents a unidirectional relation from the Volume to the VolumeTemplate.

The role of the relation should be mapped to a reference attribute of Volume. The name of the reference attribute shall be the role name.

#### 8.1.4.3.2 Roles

Name	Definition
templateId	This role (when present) represents the template identifier of the volume which created by using an existing volume template.

#### 8.1.4.3.3 Constraint

Name	Definition
—	—

### 8.1.4.4 Relation VirtualMachine-VirtualMachineImage (M)

#### 8.1.4.4.1 Definition

This represents a unidirectional relation from the VirtualMachine to the VirtualMachine Image.

The role of the relation should be mapped to a reference attribute of VirtualMachine. The name of the reference attribute should be the role name.

#### 8.1.4.4.2 Roles

Name	Definition
imageId	This role (when present) represents the image identifier of the VM which created by using a exiting virtual machine image. When creating a VM, if the parameter “os” is empty, the parameter “imageId” must be specified.

#### 8.1.4.4.3 Constraint

Name	Definition
os	The operation system of a VM. When creating a VM, if the parameter “os” is empty, the parameter “imageId” must be specified.

#### 8.1.4.5 Relation VirtualMachine-Volume (M)

##### 8.1.4.5.1 Definition

This represents a unidirectional relation from the VirtualMachine to the Volume.

The role of the relation should be mapped to a reference attribute of VirtualMachine. The name of the reference attribute shall be the role name.

##### 8.1.4.5.2 Roles

Name	Definition
volumeldList	This role (when present) represents the volume identifiers contained in the VM.

##### 8.1.4.5.3 Constraint

Name	Definition
—	—

#### 8.1.4.6 Relation VirtualMachine-PhysicalNetworkInterface (M)

##### 8.1.4.6.1 Definition

This represents a unidirectional relation from the Virtual Machine to the Physical Network Interface.

The role of the relation should be mapped to a reference attribute of Virtual Machine. The name of the reference attribute shall be the role name.

#### 8.1.4.6.2 Roles

Name	Definition
phyNetInterfaceIdList	This role (when present) represents the physical network interface identifiers contained in the VM.

#### 8.1.4.6.3 Constraint

Name	Definition
—	—

#### 8.1.4.7 Relation VirtualMachine-VirtualNetworkInterface (M)

##### 8.1.4.7.1 Definition

This represents a unidirectional relation from the Virtual Machine to Virtual Network Interface.

The role of the relation should be mapped to a reference attribute of Virtual Machine. The name of the reference attribute shall be the role name.

##### 8.1.4.7.2 Roles

Name	Definition
vNetInterfaceIdList	This role (when present) represents the virtual network interface identifiers contained in the VM. When this role is present, the VM. <b>vNetInterfaces</b> should carry the list of the vNetInterfaceId.

##### 8.1.4.7.3 Constraint

Name	Definition
phyNetInterfaceIdList	This role (when present) represents the physical network interface identifiers which the vNetInterface is mapped to.

#### 8.1.4.8 Relation VirtualNetworkInterface-PhysicalNetworkInterface (M)

##### 8.1.4.8.1 Definition

This represents a unidirectional relation from the virtual network interface to physical network interface.

The role of the relation should be mapped to a reference attribute of Virtual network interface. The name of the reference attribute shall be the role name.

##### 8.1.4.8.2 Roles

Name	Definition
phyNetworkInterfaceId	This role (when present) represents the physical network interface which the virtual network interface would be mapped.

#### 8.1.4.8.3 Constraint

Name	Definition
—	—

### 8.1.5 Information attribute definitions

#### 8.1.5.1 Definition and legal values

Attribute name	Definition	Information type/Legal values
managementDomainId	The identifier of the management domain.	Name
userLabel	The friendly name of the management domain.	String
managedResourceIdList	The list of resource Id that are managed by this management domain.	Set of name
rplId	The identifier of the RPL.	Name
sub-resourcePoolIdList	The list of child resource pool Id that included in this RPL.	Set of name
maxCapacity	The maximum number of resources this RPL can hold.	String If not provided, assume it is unlimited.
minCapacity	The minimal number of resources this RPL can hold.	String If not specified, assume it is 1.
createdTime	Date and time of the resource created.	Time
resourcePoolStatus	The current status of the resource pool.	String The values may be one of the followings: {INITIATED, CREATING, CREATED, DESTROYING, DESTROYED, READY }
templateId	The identifier of the template.	Name
templateStatus	The current status of the template.	String The values may be one of the followings: {INITIATED, CREATING, CREATED, DESTROYING,

Attribute name	Definition	Information type/Legal values
		DESTROYED, READY }
vmId	The identifier of the VM.	Name
os	The operating system running on the VM.	String
cpu	Count of CPU cores and CPU cores speed in MHz of the VM.	[Number, Number]
memory	Main memory size in MB of the VM.	Integer
disks	The name and the size in GB of local disks.	{String, Number} []
volumeldList	The list of volume Id that attached to this VM.	Set of name
imageId	The virtual machine image Id which is used by VM. If the os is empty when creating a VM, the imageId must be specified.	Name
phyNetInterfaceIdList	The physical network interface Id that attached to this VM.	Set of name
vNetInterfaceIdList	The list of virtual network interface Id that attached to this VM.	Set of name
hostname	The name of the virtualized physical host that this VM is contained in.	String
resourcePool	Resource Pool Id of this VM contained in.	Name
priority	The VM boot order or close order.	String The values may be one of the followings: {High, Medium, Low }
snapshots	The information of a snapshot that have been taken by this VM. The information of a snapshot includes snapshot ID creating time and the snapshot status.	string
vmStatus	The current status of the VM.	String The values may be one of the followings: {INITIATED,

Attribute name	Definition	Information type/Legal values
		CREATING, CREATED, DESTROYING, DESTROYED, READY, STOPPED, STOPPING, STARTING, STARTED, SUSPENDED, SUSPENDING, RESUMING, RESTARTING }
imageId	The identifier of the VM image.	Name
imageName	The name of the image, which is convenient to read.	String
volumeId	The identifier of the volume.	Name
size	The size of the volume in GBytes.	Integer
location	The location of this volume is contained.	Set of name
resourcePool	The immediate resource Pool Id of this volume contained in.	Name
volumeStatus	The current status of the template.	String The values may be one of the followings: {INITIATED, CREATING, CREATED, DESTROYING, DESTROYED, READY }
phyNetInterfaceId	The identifier of the physical network interface.	Name
macAddress	MAC address of the network interface.	string
ipAddress	IP address of the network interface.	string
netInterfaceStatus	The current status of the physical network interface or virtual network interface.	String The values may be one of the followings: {ADDED,

Attribute name	Definition	Information type/Legal values
		ACTIVATED, INACTIVATED, PASSIVE, DISABLED, }
vNetInterfaceId	The identifier of the virtual network interface.	Name
maxLimitRate	The maximum limit rate of the virtual network interface. Unit: kbytes/s	Integer
resourcePolicyId	The identifier of resource scheduling policy.	Name
resourcePolicyDescription	The description of the resource scheduling policy.	String

### 8.1.5.2 Constraints

None.

## 8.2 Interface definition

### 8.2.1 Class diagram representing interfaces

The class diagram for basic object management reuses the definition in [ITU-T Rec. M.3700]. The class diagram for alarm management IRP reuses the definition in [ITU-T Rec. M.3703], and the class diagram for performance management IRP reuses the definition in [ITU-T Rec. M.3704].

Figure 8-8 depicts the other class diagram exclude the basic object management, alarm management and performance management, including Cloud Virtualized Resource Configuration Management IRP (CVRCMIRP).

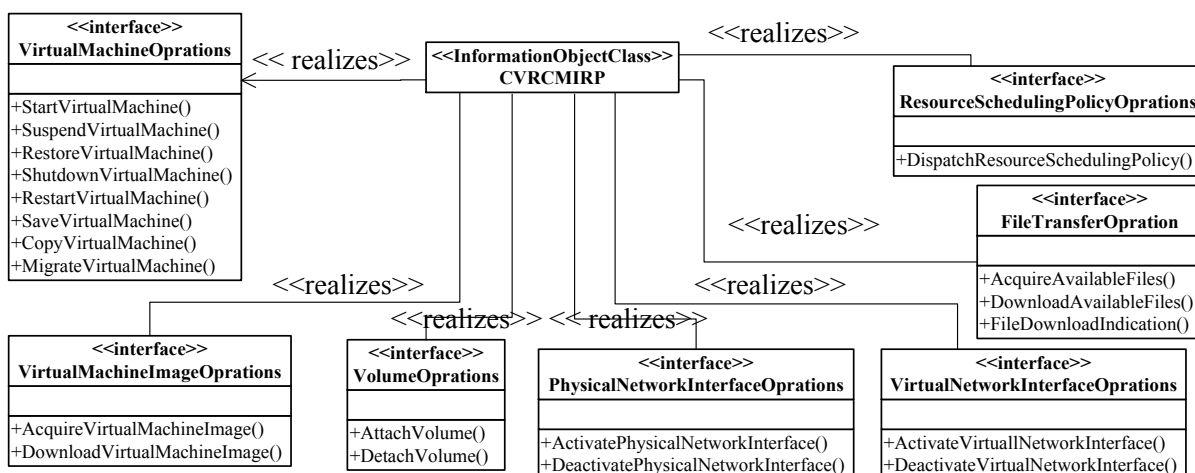


Figure 8-8 – Class diagram

### 8.2.2 Generic rules

*The following rules are relevant to all specifications. They shall simply be copied as part of the specification.*

*Rule 1: Each operation with at least one input parameter supports a pre-condition `valid_input_parameter` which indicates that all input parameters shall be valid with regard to their information type. Additionally, each such operation supports an exception `operation_failed_invalid_input_parameter` which is raised when `valid_input_parameter` is false. The exception has the same entry and exit state.*

*Rule 2: Each operation with at least one optional input parameter supports a set of pre-conditions `supported_optional_input_parameter_xxx` where "xxx" is the name of the optional input parameter and the pre-condition indicates that the operation supports the named optional input parameter. Additionally, each such operation supports an exception `operation_failed_unsupported_optional_input_parameter_xxx` which is raised when (a) the pre-condition `supported_optional_input_parameter_xxx` is false and (b) the named optional input parameter is carrying information. The exception has the same entry and exit state.*

*Rule 3: Each operation shall support a generic exception `operation_failed_internal_problem` which is raised when an internal problem occurs and that the operation cannot be completed. The exception has the same entry and exit state.*

### 8.2.3 Configuration Management Interface (M)

#### 8.2.3.1 ResourcePoolOperations interface (M)

The interface for basic object management reuses the definition in [ITU-T Rec. M.3700], including Create object, Modify object, Delete object and Retrieve object information.

When create a resource pool, the input characteristic of the object instance could be specified in the Parameter 'attributeListIn' in clause 7.3.3.3.2 of [ITU-T Rec. M.3700], such as template ID or the RPL configuration parameters, include resource pool Id, description, maximum capacity, minimum capacity and so on. The output characteristic of the object instance could be specified in the Parameter 'attributeListOut' in clause 7.3.3.3.3 of [ITU-T Rec. M.3700], such as the resource pool ID, creating time and the current state.

When modify a resource pool, the parameters which would be modified could be specified in the Parameter 'modificationList' in the clause 7.3.3.5.2 of [ITU-T Rec. M.3700], such as resource pool ID, description and so on.

#### 8.2.3.2 TemplateOperations interface (M)

The interface for basic object management reuses the definition in [ITU-T Rec. M.3700], including Create object, Modify object, Delete object and Retrieve object information.

When create a template, the input characteristic of the object instance could be specified in the Parameter 'attributeListIn' in clause 7.3.3.3.2 of [ITU-T Rec. M.3700]. The output characteristic of the object instance could be specified in the Parameter 'attributeListOut' in clause 7.3.3.3.3 of [ITU-T Rec. M.3700].

When modify a template, the parameters which would be modified could be specified in the Parameter 'modificationList' in the clause 7.3.3.5.2 of [ITU-T Rec. M.3700].



#### 8.2.3.2.1 Operation AcquireTemplate

The COSS could acquire the templates stored in the VRM in file format using some file transfer mechanisms. This operation reuses the definition in the clause 8.2.3.9.1 in File transfer management.

#### 8.2.3.2.2 Operation DownloadTemplate

The VRM could download the templates stored in the COSS in file format using some file transfer mechanisms. This operation reuses the definition in the clause 8.2.3.9.2 in File transfer management.

#### 8.2.3.3 VirtualMachineOperations interface (M)

The interface for basic object management reuses the definition in [ITU-T Rec. M.3700], including Create object, Modify object, Delete object and Retrieve object information.

When create a VM, the input characteristic of the object instance could be specified in the Parameter ‘attributeListIn’ in clause 7.3.3.3.2 of [ITU-T Rec. M.3700]. The output characteristic of the object instance could be specified in the Parameter ‘attributeListOut’ in clause 7.3.3.3.3 of [ITU-T Rec. M.3700]. The VM creation data includes resource pool ID, OS, CPU, memory, disks and physical network Interface Id and so on.

When modify a VM, the parameters which would be modified could be specified in the Parameter ‘modificationList’ in the clause 7.3.3.5.2 of [ITU-T Rec. M.3700]. The VM modification data includes CPU, memory, disks and physical network Interface Id or virtual network interface Id, priority, and so on.

#### 8.2.3.3.1 Operation StartVirtualMachine

##### 8.2.3.3.1.1 Definition

The COSS invokes this operation to start a virtual machine.

##### 8.2.3.3.1.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of a VM.

##### 8.2.3.3.1.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmStatus	M	String The values may be one of the followings: {INITIATED, CREATING, CREATED, DESTROYING, DESTROYED, READY, STOPPED, STOPPING, STARTING, STARTED, SUSPENDED, SUSPENDING, RESUMING, RESTARTING }	It specifies the start operation whether is successful. If the operation is successful, vmStatus= STARTED.

#### 8.2.3.3.1.4 Pre-condition

The specified VM exists and at the “Created” status.

#### 8.2.3.3.1.5 Post-condition

The specified VM is started, and the status of the VM is changed to “Started”.

#### 8.2.3.3.1.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented". Exit state: Entry state.

Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed". Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: "Forbidden". Exit state: Entry state.

### 8.2.3.3.2 Operation SuspendVirtualMachine

#### 8.2.3.3.2.1 Definition

The COSS invokes this operation to suspend a virtual machine which is at "Started" status.

#### 8.2.3.3.2.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of the VM.

#### 8.2.3.3.2.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmStatus	M	String The values may be one of the followings: {INITIATED, CREATING, CREATED, DESTROYING, DESTROYED, READY, STOPPED, STOPPING, STARTING, STARTED, SUSPENDED, SUSPENDING, RESUMING, RESTARTING }	It specifies the operation whether is successful. If the operation is successful, vmStatus= SUSPENDED.

#### 8.2.3.3.2.4 Pre-condition

The VM exists and the VM is at the "Started" status.

#### 8.2.3.3.2.5 Post-condition

The VM is at “Suspended” status after the operation.

#### 8.2.3.3.2.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: " Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: “Not Found”. Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: “Method Not Allowed”. Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: “Forbidden ”. Exit state: Entry state.

#### 8.2.3.3.3 Operation RestoreVirtualMachine

##### 8.2.3.3.3.1 Definition

The COSS invokes this operation to restore a VM which is at “Suspended” status.

##### 8.2.3.3.3.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of the VM.

##### 8.2.3.3.3.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmStatus	M	String The values may be one of the followings: {INITIATED, CREATING, CREATED, DESTROYING, DESTROYED, READY, STOPPED,	It specifies the operation whether is successful. If the operation is successful, vmStatus= STARTED.

		STOPPING, STARTING, STARTED, SUSPENDED, SUSPENDING, RESUMING, RESTARTING }	
--	--	---	--

#### 8.2.3.3.3.4 Pre-condition

The VM exists and is at the “Suspended” status.

#### 8.2.3.3.3.5 Post-condition

The VM is at “Started” status after the operation.

#### 8.2.3.3.3.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: " Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: “Not Found”. Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: “Method Not Allowed”. Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: “Forbidden ”. Exit state: Entry state.

#### 8.2.3.3.4 Operation ShutdownVirtualMachine

##### 8.2.3.3.4.1 Definition

The COSS invokes this operation to shut down a VM which is at is “started” status.

##### 8.2.3.3.4.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of the VM.

##### 8.2.3.3.4.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmStatus	M	<p>String</p> <p>The values may be one of the followings: {INITIATED, CREATING, CREATED, DESTROYING, DESTROYED, READY, STOPPED, STOPPING, STARTING, STARTED, SUSPENDED, SUSPENDING, RESUMING, RESTARTING }</p>	<p>It specifies the operation whether is successful.</p> <p>If the operation is successful, vmStatus= STOPPED.</p>

#### 8.2.3.3.4.4 Pre-condition

The VM exists and is at the “Started” status.

#### 8.2.3.3.4.5 Post-condition

The VM is at “Stopped” status after the operation.

#### 8.2.3.3.4.6 Exceptions

Name	Definition
Not Implemented	<p>Condition: The server does not(currently) support the functionality required to fulfil the request.</p> <p>Returned information: " Not Implemented".</p> <p>Exit state: Entry state.</p>
Not Found	<p>Condition: The request specified an Id of a resource that does not exist.</p> <p>Returned information: “Not Found”.</p> <p>Exit state: Entry state.</p>
Method Not Allowed	<p>Condition: The operation verb specified in the request is not supported.</p> <p>Returned information: “Method Not Allowed”.</p> <p>Exit state: Entry state.</p>
Forbidden	<p>Condition: The server recognized the credentials, but you do not possess authorization to perform this request.</p> <p>Returned information: “Forbidden ”.</p> <p>Exit state: Entry state.</p>

### 8.2.3.3.5 Operation RestartVirtualMachine

#### 8.2.3.3.5.1 Definition

The COSS invokes this operation to restart a VM which is at the “Stopped” status.

#### 8.2.3.3.5.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of the VM.

#### 8.2.3.3.5.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmStatus	M	String The values may be one of the followings: {INITIATED, CREATING, CREATED, DESTROYING, DESTROYED, READY, STOPPED, STOPPING, STARTING, STARTED, SUSPENDED, SUSPENDING, RESUMING, RESTARTING }	It specifies the operation whether is successful. If the operation is successful, vmStatus= STARTED.

#### 8.2.3.3.5.4 Pre-condition

The VM exists and is at the “Stopped” status.

#### 8.2.3.3.5.5 Post-condition

The VM is at “Started” status after the operation.

#### 8.2.3.3.5.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: " Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not

	exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed". Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: "Forbidden". Exit state: Entry state.

### 8.2.3.3.6 Operation SaveVirtualMachine

#### 8.2.3.3.6.1 Definition

The COSS invokes this operation to save a VM, after the operation, a new snapshot is created and the COSS receives the snapshot information as a response.

#### 8.2.3.3.6.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of the VM.

#### 8.2.3.3.6.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
snapshotId	M	String	It specifies the identifier of the snapshot.
createdTime	M	Time	The time of the snapshot is created.
snapshotStatus	M	String The values may be one of the followings: {INITIATED, CREATING, CREATED, DESTROYING, DESTROYED, READY }	The current status of the snapshot. If the operation is successful, snapshotStatus = CREATED.

#### 8.2.3.3.6.4 Pre-condition

The VM should be suspended, and at the 'Suspended' status.

#### 8.2.3.3.6.5 Post-condition

A snapshot created and the snapshot information is returned as a response.



#### 8.2.3.3.6.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: " Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed". Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: "Forbidden ". Exit state: Entry state.

#### 8.2.3.3.7 Operation CopyVirtualMachine

##### 8.2.3.3.7.1 Definition

The COSS invokes this operation to copy the VM, after the operation, a new VM is created and the new VM ID is feed back to the COSS as a response

##### 8.2.3.3.7.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of a VM.
newVMId	M	Name	It specifies the identifier of the new VM.
physicalServerId	M	Name	It specifies the location of the new VM.

##### 8.2.3.3.7.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
newVMId	M	Name	It specifies the identifier of the new VM.

##### 8.2.3.3.7.4 Pre-condition

The specified VM, as well as the physical server which is used to store the new VM should exist.

##### 8.2.3.3.7.5 Post-condition

The VM is copied successfully, and the new VM is stored to the specified physical server.

#### 8.2.3.3.7.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: " Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed". Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: "Forbidden ". Exit state: Entry state.

#### 8.2.3.3.8 Operation MigrationVirtualMachine

##### 8.2.3.3.8.1 Definition

The VM is migrated from one physical server to another physical server.

##### 8.2.3.3.8.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of the VM.
sourceLocation	M	String	The location of the VM before migration.
destinationLocation	M	String	The location of the VM which should be migrated to.

##### 8.2.3.3.8.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
response	M	Boolean	It specifies the migration operation whether is successful.

##### 8.2.3.3.8.4 Pre-condition

The specified VM, as well as the physical server which is used to store the migrated VM should exist.

##### 8.2.3.3.8.5 Post-condition

The VM is migrated to the destination physical server.

#### 8.2.3.3.8.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: " Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed". Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: "Forbidden ". Exit state: Entry state.

#### 8.2.3.4 VirtualMachineImageOperations interface (M)

The interface for basic object management reuses the definition in [ITU-T Rec. M.3700], including Create object, Delete object and Retrieve object information.

When create a VM image, the input characteristic of the object instance could be specified in the Parameter 'attributeListIn' in clause 7.3.3.3.2. The output characteristic of the object instance could be specified in the Parameter 'attributeListOut' in clause 7.3.3.3.3. The VM image creation data includes the vmID and the image name which will be generated and so on.

When modify a VM image, the parameters which would be modified could be specified in the Parameter 'modificationList' in the clause 7.3.3.5.2. The VM image query data includes image ID, image name, os and creating time.

##### 8.2.3.4.1 Operation AcquireVirtualMachineImage

The COSS could acquire the virtual machine images stored in the VRM in file format using some file transfer mechanisms. This operation reuses the definition in the clause 8.2.3.9.1 in File transfer management.

##### 8.2.3.4.2 Operation DownloadVirtualMachineImage

The VRM could download the virtual machine images stored in the COSS in file format using some file transfer mechanisms. This operation reuses the definition in the clause 8.2.3.9.2 in File transfer management.

#### 8.2.3.5 VolumeOperations interface (M)

The interface for basic object management reuses the definition in [ITU-T Rec. M.3700], including Create object, Modify object, Delete object and Retrieve object information.

When create a volume, the input characteristic of the object instance could be specified in the Parameter 'attributeListIn' in clause 7.3.3.3.2. The output characteristic of the object instance could be specified in the Parameter 'attributeListOut' in clause 7.3.3.3.3. The volume creation data includes resourcePoolID, volume size and Location and so on.

When modify a volume, the parameters which would be modified could be specified in the Parameter 'modificationList' in the clause 7.3.3.5.2. The volume modification data includes description, size and so on.

### 8.2.3.5.1 Operation AttachVolume

#### 8.2.3.5.1.1 Definition

The COSS invokes this operation to attach the volume to a specified VM.

#### 8.2.3.5.1.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of the VM that the volume would be attached to.
volumeId	M	Name	It specifies the identifier of the volume.

#### 8.2.3.5.1.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
response	M	Boolean	It specifies the operation whether is successful.

#### 8.2.3.5.1.4 Pre-condition

The specified VM and the volume should exist.

#### 8.2.3.5.1.5 Post-condition

None.

#### 8.2.3.5.1.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: " Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported.

	Returned information: "Method Not Allowed". Exit state: Entry state.
No Content	Condition: The request has been processed and no content was returned. Returned information: "No Content ". Exit state: Entry state.

### 8.2.3.5.2 Operation DetachVolume

#### 8.2.3.5.2.1 Definition

The COSS invokes this operation to detach the volume from the specified VM.

#### 8.2.3.5.2.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
vmId	M	Name	It specifies the identifier of a VM that the volume would be detached from.
volumeId	M	Name	It specifies the identifier of the volume.

#### 8.2.3.5.2.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
response	M	Boolean	It specifies the reduce operation whether is successful.

#### 8.2.3.5.2.4 Pre-condition

The specified VM and the volume should exist.

#### 8.2.3.5.2.5 Post-condition

None.

#### 8.2.3.5.2.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed". Exit state: Entry state.

No Content	Condition: The request has been processed and no content was returned. Returned information: "No Content ". Exit state: Entry state.
------------	--

#### 8.2.3.6 PhysicalNetworkInterfaceOperations interface (M)

The interface for basic object management reuses the definition in [ITU-T Rec. M.3700], including Create object, Modify object, Delete object and Retrieve object information.

When create a physical network interface, the input characteristic of the object instance could be specified in the Parameter 'attributeListIn' in clause 7.3.3.3.2. The output characteristic of the object instance could be specified in the Parameter 'attributeListOut' in clause 7.3.3.3.3. The physical network interface creation data includes IP address, MAC address and so on.

When modify a physical network interface, the parameters which would be modified could be specified in the Parameter 'modificationList' in the clause 7.3.3.5.2. The physical network interface modification data includes physical network interface ID, IP Address and so on.

##### 8.2.3.6.1 Operation ActivatePhysicalNetworkInterface

###### 8.2.3.6.1.1 Definition

The COSS invokes this operation to activate the physical network interface.

###### 8.2.3.6.1.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
phyNetworkInterface Id	M	Name	It specifies the identifier of the physical network interface.

###### 8.2.3.6.1.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
netInterfaceStatus	M	String The values may be one of the followings: {ADDED, ACTIVATED, INACTIVATED, PASSIVE, DISABLED, }	The current status of the physical network interface. If the operation is successful, netInterfaceStatus= ACTIVATED.

###### 8.2.3.6.1.4 Pre-condition

None.

###### 8.2.3.6.1.5 Post-condition

The specified physical network interface is activated, and at the "Activated" status after the operation.

#### 8.2.3.6.1.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed". Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: "Forbidden". Exit state: Entry state.

#### 8.2.3.6.2 Operation DeactivatePhysicalNetworkInterface

##### 8.2.3.6.2.1 Definition

The COSS invokes this operation to deactivate the physical network interface.

##### 8.2.3.6.2.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
phyNetworkInterface Id	M	Name	It specifies the identifier of the physical network interface.

##### 8.2.3.6.2.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
netInterfaceStatus	M	String The values may be one of the followings: {ADDED, ACTIVATED, INACTIVATED, PASSIVE, DISABLED, }	The current status of the physical network interface. If the operation is successful, netInterfaceStatus= INACTIVATED.

##### 8.2.3.6.2.4 Pre-condition

The physical network interface exists and is at the "Activated" status.

##### 8.2.3.6.2.5 Post-condition

The physical network interface is deactivated and is at the “Deactivated” status.

#### 8.2.3.6.2.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: “Not Found”. Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: “Method Not Allowed”. Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: “Forbidden ”. Exit state: Entry state.

#### 8.2.3.7 VirtualNetworkInterfaceOperations interface (M)

The interface for basic object management reuses the definition in [ITU-T Rec. M.3700], including Create object, Modify object, Delete object and Retrieve object information.

When create a virtual network interface, the input characteristic of the object instance could be specified in the Parameter ‘attributeListIn’ in clause 7.3.3.3.2. The output characteristic of the object instance could be specified in the Parameter ‘attributeListOut’ in clause 7.3.3.3.3. The virtual network interface creation data includes physical network interface ID, VM ID, MAC address and maxLimitRate and so on.

When modify a virtual network interface, the parameters which would be modified could be specified in the Parameter ‘modificationList’ in the clause 7.3.3.5.2. The virtual network interface modification data includes physical network interface ID, MAC address and maxLimitRate and so on.

#### 8.2.3.7.1 Operation ActivateVirtualNetworkInterface

##### 8.2.3.7.1.1 Definition

The COSS invokes this operation to activate the virtual network interface.

##### 8.2.3.7.1.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
virNetworkInterface Id	M	Name	It specifies the identifier of the virtual network interface.



#### 8.2.3.7.1.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
netInterfaceStatus	M	<code>String</code> The values may be one of the followings: {CREATING, CREATED, ACTIVATED, INACTIVATED, PASSIVE, DISABLED, }	The current status of the virtual network interface. If the operation is successful, netInterfaceStatus= ACTIVATED.

#### 8.2.3.7.1.4 Pre-condition

None.

#### 8.2.3.7.1.5 Post-condition

The specified virtual network interface is activated, and is at the “Activated” status.

#### 8.2.3.7.1.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: “Not Found”. Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: “Method Not Allowed”. Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: “Forbidden ”. Exit state: Entry state.

#### 8.2.3.7.2 Operation DeactivateVirtualNetworkInterface

##### 8.2.3.7.2.1 Definition

The COSS invokes this operation to deactivate the virtual network interface.

##### 8.2.3.7.2.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
virNetworkInterfaceId	M	Name	It specifies the identifier of the virtual network interface.

#### 8.2.3.7.2.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
netInterfaceStatus	M	String The values may be one of the followings: {CREATING, CREATED, ACTIVATED, INACTIVATED, PASSIVE, DISABLED, }	The current status of the virtual network interface. If the operation is successful, netInterfaceStatus=INACTIVATED.

#### 8.2.3.7.2.4 Pre-condition

The virtual network interface exists and is at the “Activated” status.

#### 8.2.3.7.2.5 Post-condition

The virtual network interface is deactivated and is at the “Deactivated” status.

#### 8.2.3.7.2.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: “Not Found”. Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: “Method Not Allowed”. Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: “Forbidden ”. Exit state: Entry state.

### 8.2.3.8 ResourceSchedulingPolicyManagementOperations interface (M)

The interface for basic object management reuses the definition in [ITU-T Rec. M.3700], including Retrieve object information. The RSP query data includes resource policy ID and resource policy description and so on.

#### 8.2.3.8.1 Operation DispatchResourceSchedulingPolicy

##### 8.2.3.8.1.1 Definion

The COSS invokes this operation to dispatch the resource scheduling policy.

##### 8.2.3.8.1.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
resourcePolicyInformation	M	String The list of the following informations: {resourcePolicyId, resource PolicyDescription }.	It specifies the resource scheduling policy information.

##### 8.2.3.8.1.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
response	M	Boolean	It specifies the issue operation whether is successful.

##### 8.2.3.8.1.4 Pre-condition

None.

##### 8.2.3.8.1.5 Post-condition

None.

##### 8.2.3.8.1.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed". Exit state: Entry state.
Forbidden	Condition: The server recognized the credentials, but you do not possess authorization to perform this request. Returned information: "Forbidden ". Exit state: Entry state.

### 8.2.3.9 File Transfer Management interface (M)

#### 8.2.3.9.1 Operation AcquireAvailableFiles

##### 8.2.3.9.1.1 Definition

The COSS invokes this operation to acquire the file(s), such as template file(s) or virtual machine image file(s), and so on.

##### 8.2.3.9.1.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
objectType	M	string	It specifies the acquire object type, includes template file(s) or image file(s).
objectIdList	M	Set of Name	It specifies the acquire object identifiers.

##### 8.2.3.9.1.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
response	M	Boolean	It specifies the acquire operation whether is invoked successfully.

##### 8.2.3.9.1.4 Pre-condition

None.

##### 8.2.3.9.1.5 Post-condition

The VRM starts a file preparation task on the request of the COSS. When the preparation is finished, the VRM will send a "NotifyFileTransferReady" notification to the COSS. When the preparation is failed, the VRM will send a "NotifyFilePreparationError" notification to the COSS.

##### 8.2.3.9.1.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed".

	Exit state: Entry state.
No Content	Condition: The request has been processed and no content was returned. Returned information: "No Content ". Exit state: Entry state.

### 8.2.3.9.2 Operation DownloadAvailableFiles

#### 8.2.3.9.2.1 Definition

The VRM invokes this operation to download the file(s), include template file(s) or image file(s) and so on.

#### 8.2.3.9.2.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
objectType	M	string	It specifies the downloaded object type, includes template file(s) or image file(s).
objectIdList	M	Set of Name	It specifies the downloaded object identifiers.

#### 8.2.3.9.2.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
response	M	Boolean	It specifies the download operation whether is invoked successfully.

#### 8.2.3.9.2.4 Pre-condition

None.

#### 8.2.3.9.2.5 Post-condition

The COSS starts a file preparation task on the request of the VRM. When the preparation is finished, the COSS will invoke FileDownloadIndication to indicate the VRM to download the files.

#### 8.2.3.9.2.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented". Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
Method Not Allowed	Condition: The operation verb specified in the request is not supported. Returned information: "Method Not Allowed".

	Exit state: Entry state.
No Content	Condition: The request has been processed and no content was returned. Returned information: "No Content ". Exit state: Entry state.

### 8.2.3.9.3 Operation FileDownloadIndication

#### 8.2.3.9.3.1 Definion

The COSS invokes this operation to indicate the VRM to download the files.

#### 8.2.3.9.3.2 Input parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
fileInfoList	M	FileInfoListType ::= SET OF SEQUENCE { fileDirectory String, fileInfoList SET OF SEQUENCE { fileName String, fileSize INTEGER, fileCompression String, fileCreationTime GeneralizedTime, fileDeletionTime GeneralizedTime } }	All information pertaining to the files are provided in this parameter.

#### 8.2.3.9.3.3 Output parameters

Parameter name	Support qualifier	Information type/Legal values	Comment
response	M	Boolean	It specifies the issue operation whether is successful.

#### 8.2.3.9.3.4 Pre-condition

The fileInfoList is valid.

#### 8.2.3.9.3.5 Post-condition

The downloaded file exists.

#### 8.2.3.9.3.6 Exceptions

Name	Definition
Not Implemented	Condition: The server does not(currently) support the functionality required to fulfil the request. Returned information: "Not Implemented".

	Exit state: Entry state.
Not Found	Condition: The request specified an Id of a resource that does not exist. Returned information: "Not Found". Exit state: Entry state.
No Content	Condition: The request has been processed and no content was returned. Returned information: "No Content". Exit state: Entry state.

#### 8.2.3.9.4 Notification notifyFileTransferReady

##### 8.2.3.9.4.1 Definition

This notification is used to inform the COSS that the requested file(s) is prepared and ready for transfer.

##### 8.2.3.9.4.2 Input parameters

Parameter name	Qualifiers	Matching information/ Information type/ Legal values	Comment
transferId	M	INTEGER	This is the transfer identifier for this file transfer transaction.
fileInfoList	M	FileInfoListType	All information pertaining to the files are provided in this parameter. See clause 8.2.3.9.1.2 for details.
ipAddress	O	String	The IP address of the host machine where the files are located.
userName	O	String	The user name to be used in FTP.
password	O	String	The password to be used in FTP.

##### 8.2.3.9.4.3 Triggering event

###### 8.2.3.9.4.3.1 From state

FilePreparationStarted.

Assertion name	Definition
FilePreparationStarted	Once receiving the template or image acquire request of a COSS, the VRM will start preparing the template file(s) or image file(s).

###### 8.2.3.9.4.3.2 To state

FilePreparationFinishedSuccessfully.

Assertion name	Definition
FilePreparationFinished Successfully	The requested file(s) is prepared successfully.

#### 8.2.3.9.5 Notification notifyFilePreparationError

##### 8.2.3.9.5.1 Definition

This notification is used to inform the COSS that an error occurred while the VRM was preparing the requested file(s).

##### 8.2.3.9.5.2 Input parameters

Parameter name	Qualifiers	Matching information/ Information type/ Legal values	Comment
transferId	M	INTEGER	This is the transfer identifier for this file transfer transaction.
reason	M	String	This indicates error reason why file transfer preparation failed.
perceivedSeverity	M	ENUMERATED {major, minor, warning} / "major": the severity is high "minor": the severity is low "warning": the severity is very low	This indicates the severity of the error, can be one of the following: major, minor, warning.
additionalText	O	String	This parameter may be used to specify additional information about this error.

##### 8.2.3.9.5.3 Triggering event

###### 8.2.3.9.5.3.1 From state

FilePreparationStarted.

Assertion name	Definition
FilePreparationStarted	Once receiving the template or image acquire request of a COSS, the VRM will start preparing the template file(s) or image file(s).

###### 8.2.3.9.5.3.2 To state

FilePreparationFinishedUnsuccessfully.

Assertion name	Definition
FilePreparationFinished	The requested file(s) is not prepared successfully, and some errors



Unsuccessfully	occurred during the preparation procedure.
----------------	--

#### 8.2.4 FaultManagementOperations interface (M)

The fault management operation reuses the definition in the [ITU-T Rec M.3703].

#### 8.2.5 PerformanceManagementOperations interface (M)

The performance management operation reuses the definition in the [ITU-T Rec M.3704].

The performance parameters of the virtual resource in the cloud computing environment are as follows:

Parameter Name	Comment	Information type/Legal values
CPU utilization	The CPU utilization ratio of the VM.	Float The value should be (0..100).
memory utilization	The memory utilization ratio of the VM.	Float The value should be (0..100).
disk utilization	The disk utilization ratio of the VM.	Float The value should be (0..100).
storage utilization	The storage utilization of the volume.	Float The value should be (0..100).
I/O traffic	The input or output traffic of the volume.	Integer Unit: Mbytes/s
network traffic	The network traffic of the network interface.	Integer Unit: Mbytes
bandwidth	The bandwidth of the network interface.	Integer Unit: Mbytes/s
packet loss ratio	The packet loss ratio of the network interface.	Float The value should be (0..100).
time delay	The time delay of the network interface.	Integer Unit: ms

## Bibliography

The following documents have been used in the production of this Recommendation:

- [b-DMTF DSP2027] DMTF DSP2027(2012), *Cloud Infrastructure Management Interface(CIMI)Primer V1.0.1 2012-9-12*
- [b-DMTF DSP0263] DMTF DSP0263 (2012), *Cloud Infrastructure Management Interface Model and REST Interface Over HTTP-based Protocol V1.0.1 2012-9-12*
- [b-IETF Draft] IETF DRAFT (2011),*Virtual Resource Management in Cloud.*
- [b- OGF GFD-P-R.183] OGF GFD-P-R.183 (2011), *Open Cloud Computing Interface – Core*
- [b- OGF GFD-P-R.184] OGF GFD-P-R.184 (2011), *Open Cloud Computing Interface – Infrastructure*
- [b- OGF GFD-P-R.185] OGF GFD-P-R.185 (2011), *Open Cloud Computing Interface –RESTful HTTP Rendering*
- [b- ITU-T DRAFT] ITU-T DRAFT (2012), *End to End Cloud Computing Resources management requirements (Y.e2eccrmr) .*

|

---