International Telecommunication Union

ITU-T

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

FG Cloud TR

Version 1.0
(02/2012)

Focus Group on Cloud Computing
Technical Report

Part 2: Functional requirements and reference architecture

FOREWORD

The procedures for establishment of focus groups are defined in Recommendation ITU-T A.7. The ITU-T Focus Group on Cloud Computing (FG Cloud) was established further to ITU-T TSAG agreement at its meeting in Geneva, 8-11 February 2010, followed by ITU-T study group and membership consultation.

Even though focus groups have a parent organization, they are organized independently from the usual operating procedures of the ITU, and are financially independent. Texts approved by focus groups (including Technical Reports) do not have the same status as ITU-T Recommendations.

INTELLECTUAL PROPERTY RIGHTS

The ITU draws attention to the possibility that the practice or implementation of this Technical Report may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU-T Focus Group participants or others outside of the Technical Report development process.

.

© ITU 2012

All rights reserved. No part of this publication may be reproduced, by any means whatsoever, without the prior written permission of ITU.

Table of Contents

1. Scope

The scope of this Technical Report is to define the functional requirements and reference architecture of cloud computing, which includes the functional architecture, functional layers and blocks.

2. Definitions

2.1 Terms defined elsewhere:

This Technical Report uses the following terms defined elsewhere:

This Technical Report uses the following terms defined in Part 1 of the Focus Group on Cloud Computing Technical Report:

- Cloud computing
- Cloud service
- Cloud service partner (CSN)
- Cloud service provider (CSP)
- Cloud service user (CSU)
- Community cloud
- Communication as a service (CaaS)
- Desktop as a service (DaaS)
- Hybrid cloud
- Infrastructure as a service (IaaS)
- Inter-cloud computing
- Inter-cloud federation
- Inter-cloud peering
- Inter-cloud service broker (ISB)
- Network as a service (NaaS)
- Platform as a service (PaaS)
- Private cloud
- Public cloud
- Resources
- Service delivery platform as a service (SDPaaS)
- Software as a service (SaaS).

2.2 Acronyms

This Technical Report uses the following acronyms:

API	Application Programming Interface
BPEL	Business Process Execution Language
BPMN	Business Process Modelling Notation
BSS	Business Support System
CaaS	Communication as a Service
CPU	Central Processing Unit
CSN	Cloud Service Partner
CSP	Cloud Service Provider
CSR	Cloud Service Requester
CSU	Cloud Service User
DaaS	Desktop as a Service
DSaaS	Data Storage as a Service
HTML	Hyper Text Mark-up Language
HTTP	Hyper Text Transfer Protocol
IaaS	Infrastructure as a Service
ISB	Inter-Cloud Service Broker
NaaS	Network as a Service
OAM&P	Operations Administration Maintenance and Provisioning
OS	Operating System
OSS	Operations Support System
OVF	Open Virtual Format
PaaS	Platform as a Service
POD	Performance-Optimized Data centre
REST	Representational State Transfer
QoS	Quality of Service
SaaS	Software as a Service
SDK	Software Development Kit
SDPaaS	Service Delivery Platform as a Service
SLA	Service Level Agreement
SOAP	Simple Object Access Protocol
VM	Virtual Machine
VPN	Virtual Private Network
XML	Extensible Mark-up Language

3. Cloud architecture requirements

The cloud architecture must meet several requirements to enable sustained innovation and development of cloud services. With multiple stakeholders involved, the cloud architecture must be flexible to fit the needs of infrastructure CSPs, CSPs and service resellers. The following high-level requirements are broadly envisioned for the cloud architecture.

- Cloud deployments will have to support many standards within the same cloud infrastructure, e.g. in terms of resource allocation, orchestration, or CSU access. The cloud architecture must allow and support the evolution of these standards, without requiring disruptive infrastructure changes from the CSP perspective.
- A cloud CSP must be able to support multiple standards within the same architecture and migrate to a newer standard if they so wish, without having to change everything in the CSP network or lose the existing customer base.
- Broadband access is fundamental in making cloud services viable. The cloud architecture may benefit from integration with the support of network resource reservation and guaranteed quality of service capabilities through the network over which services are delivered. Without the network access guaranteeing bounded delay, jitter, bandwidth, and reliability, the cloud experience for CSUs may be worse than the intranet experience.
- The cloud architecture must enable multiple deployment models, cloud service categories and use cases, some currently known and others to be envisioned in the future. Currently known cloud service categories include IaaS, PaaS, SaaS, CaaS, and NaaS, and it is possible that these will also co-exist in the same cloud deployment. The same architecture must allow a cloud service provider to provide either all, or a subset, of these services.
- For private and hybrid cloud operations, cloud services must appear like intranet services. This means a user must be able to access resources using the same domain names as on the intranet. Hosts and resources that have been migrated from private to public clouds should be accessed transparently where they are being currently hosted.
- The cloud architecture must enable early detection, diagnosis and fixing of infrastructure or service-related problems.
- CSUs must be able to (request) audit CSP's services and get assurance that the agreed-upon SLAs are being complied with. To that end, the cloud architecture must enable, among others, service-level monitoring of resources allocated to a user and generate SLA compliance reports.
- Cloud resource allocations should be invisible to the CSUs, even though services are visible. A CSP may choose to expose service-operation details without having to share cloud internal infrastructure allocation and provisioning details. This is important for a CSP for security and business reasons.
- Users consuming cloud services must be able to control cloud resource access to the CSP transparently, and enable IT procedures to work without compromise in legal or organizational mandates. This includes, for instance, the ability to dynamically add or remove a user from access to a cloud without CSP intervention.
- The cloud architecture must enable intranet-level security on the network. This *may* include access records, activity reports, session monitoring, and packet inspections on the network. It must *also* include firewalling, access control and malicious attack detection and prevention. Prevention of one user disrupting others' services is paramount.

- The cloud architecture must support cloud resource mobility which includes virtual machine mobility within a POD or data centre, between PODs or data centres within the same CSP's infrastructure, or between different CSPs' infrastructures, or from a CSU to a CSP.
- Resource mobility depends upon being able to treat an entire network as a single entity, which implies the need of the cloud architecture to scale. With a huge number of computing, storage and network resources, and an even greater number of virtualized resources, the cloud architecture must have scalability as a primary requirement.
- Naming extensions are necessary to meet cloud needs. Users who move their private resources into the cloud may need to access their resources by the same names as they did prior to those resources being migrated. Since hosts are associated with user's domain names, it is necessary to translate the user's domain names into cloud names.
- Cloud-service deployment needs to be automated in order to support scalable resource operations, including configuration, provisioning, charging, etc. In a typical scenario, a user would want to specify the computing, storage and virtual machine (VM) resources needed, as well as the network resources. This includes how the network resources should be reserved, configured and managed during lifetime for optimized connectivity between the distributed computing, storage and VM resources, and finally retired. [For more details about cloud resource management, refer to the FG CC Cloud Resource Management Technical Report as well as the FG Cloud Infrastructure Technical Report.]

4. Cloud computing reference architecture - Layering framework

4.1 Cloud architecture layering

Figure 1 shows the cloud-layering framework that is defining the layers of the cloud functional architecture. These functional layers in the architecture are derived by grouping cloud related functions.

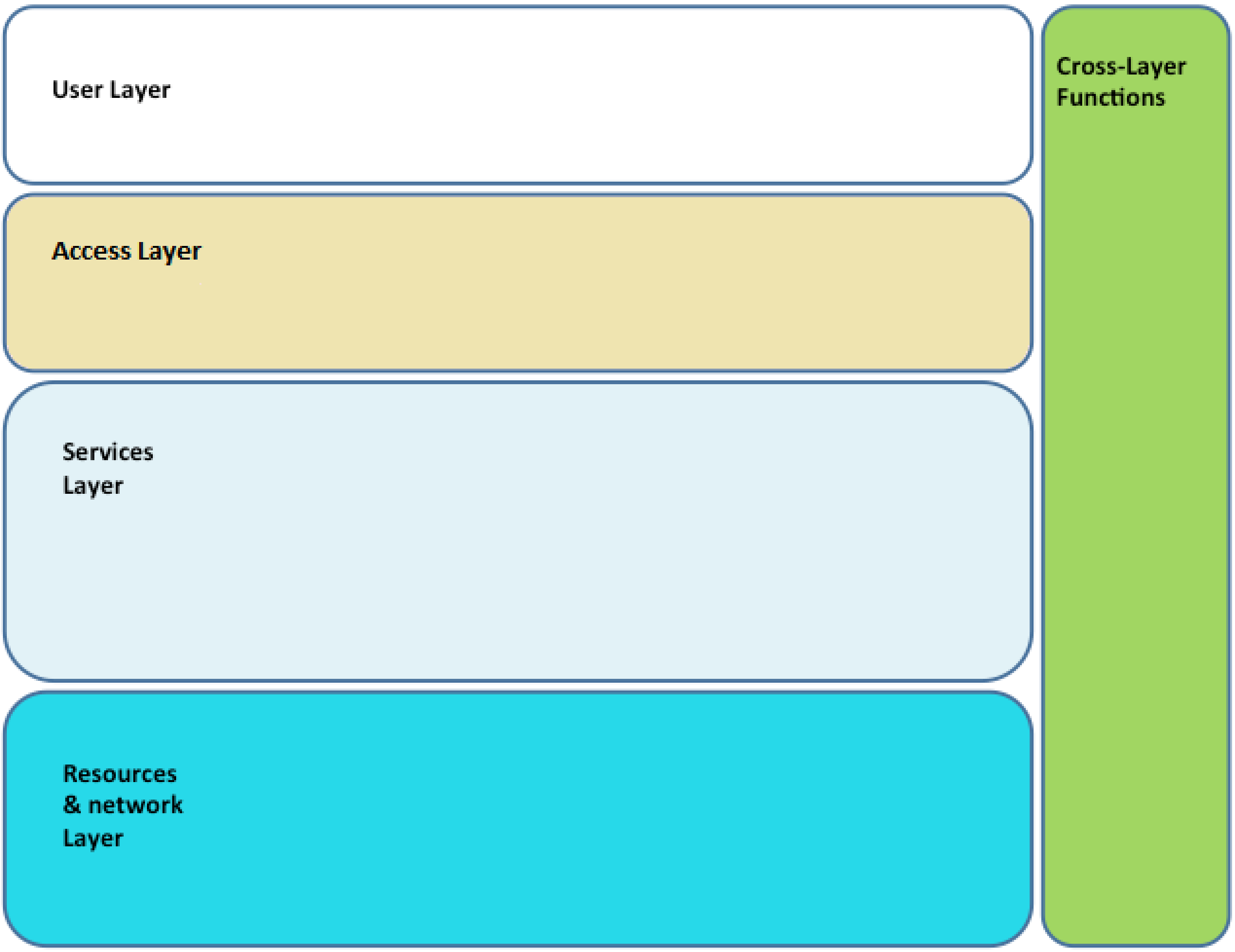

Figure 1 – Cloud-layering framework

The following sub-clauses give a description of each layer.

4.2 User layer

The user layer performs interaction between the CSU (CSU) and the cloud infrastructure. The user layer is used to set up a secure mechanism with the cloud, to send cloud service requests to the cloud, and receive cloud services from the cloud, perform cloud service access, and administrate and monitor cloud resources. When the cloud receives service requests, it orchestrates its own resources and/or other clouds' resources (if other clouds' resources are received via the inter-cloud function) and provides back-cloud services through the user layer.

The user layer is where the CSU resides. The CSU is an essential actor of the cloud ecosystem. It represents the entities that use the services offered by the CSP (CSP).

NOTE – A CSU may use a service directory from a CSP and, after setting up an appropriate contract, would use the services accordingly.

CSUs demand SLAs. SLAs govern the characteristics of the services provided by CSPs. NOTE - SLAs govern many aspects of the services provided, for example, quality of service, governance, security, performance and data portability.

The CSU typically consumes services in five main models:

1. CaaS
2. SaaS
3. PaaS
4. IaaS
5. NaaS

Please see ecosystem document for the definition of these five service models.

4.3 Access layer

The access layer provides a common interface for both manual and automated cloud service capabilities and service consumption. The access layer accepts CSU's (user) and/or CSN's (partner) and/or other CSP's cloud service consumption requests using cloud APIs to access CSPs' services and resources.

4.4 Services layer

The services layer is where the CSP orchestrates and exposes services of the five cloud service categories. It is the entity responsible for making a service available to interested parties. The cloud service layer manages the cloud infrastructure required for providing the services, runs the software that implements the services, and arranges to deliver the cloud services to the CSU through network access

For Saas or CaaS, the services layer deploys, configures, maintains and updates the operation of the software or communication applications on a cloud infrastructure so that the services are provisioned at the expected service levels to the layers above.

For PaaS, the services layer manages the cloud infrastructure for the platform and runs the software that provides the components of the platform, such as runtime software execution stack, databases, and other middleware components. The CSP of PaaS services typically also supports the development, deployment and management process of the CSUs of PaaS services by providing tools such as integrated development environments (IDEs), software development kits (SDKs), deployment and management tools. The CSUs of PaaS services have control over the applications and possibly some of the hosting environment settings, but have no, or limited access, to the underlying infrastructure.

For IaaS, the services layer controls the resources underlying the service, including the servers, networks, storage and hosting infrastructure. The services layer then runs the software necessary to makes the resources available to the CSUs of IaaS services through a set of service interfaces and resource abstractions, such as virtual machines and virtual network interfaces.

For NaaS, the services layer uses the underlying networking and transport facilities to deliver networking services in a cloud environment. Examples are network virtualization services like VPNs (see ecosystems NaaS definition).

4.5 Resources and network layer

The resources and network layer is where the physical resources reside. It includes equipment typically used in a data centre, such as servers, networking switches, storage, etc. It also represents and houses the cloud core transport network functionality which is required to provide underlying network connectivity between the CSP and the CSUs. CSUs can obtain cloud services through different network access devices, such as desktop computers, laptops, mobile phones, and mobile Internet devices. The connectivity to cloud services is normally provided by CSPs. Note that for a CSP to provide services consistent with the level of SLAs offered to the CSUs, it may require dedicated and/or secure connections between CSUs and CSPs.

4.6 Cross-layer functions

The cross-layer functions perform overall system management (i.e., operations, administration, maintenance and provisioning (OAM&P)) and monitoring, and provide secure mechanisms.

Secure mechanisms provide protections from threats to cloud computing services and infrastructure. These threats are defined in section 6 of the cloud security output document. Secure mechanisms are required by both CSUs and CSPs to create a secure cloud environment.

5. Cloud computing reference architecture – Functional blocks

Based on the layering framework described in clause 4, Figure 2 below shows the major functional blocks of the cloud computing reference architecture. It is recognized that CSPs will decide which functional blocks are appropriate to their business, and how the chosen functional blocks are implemented.

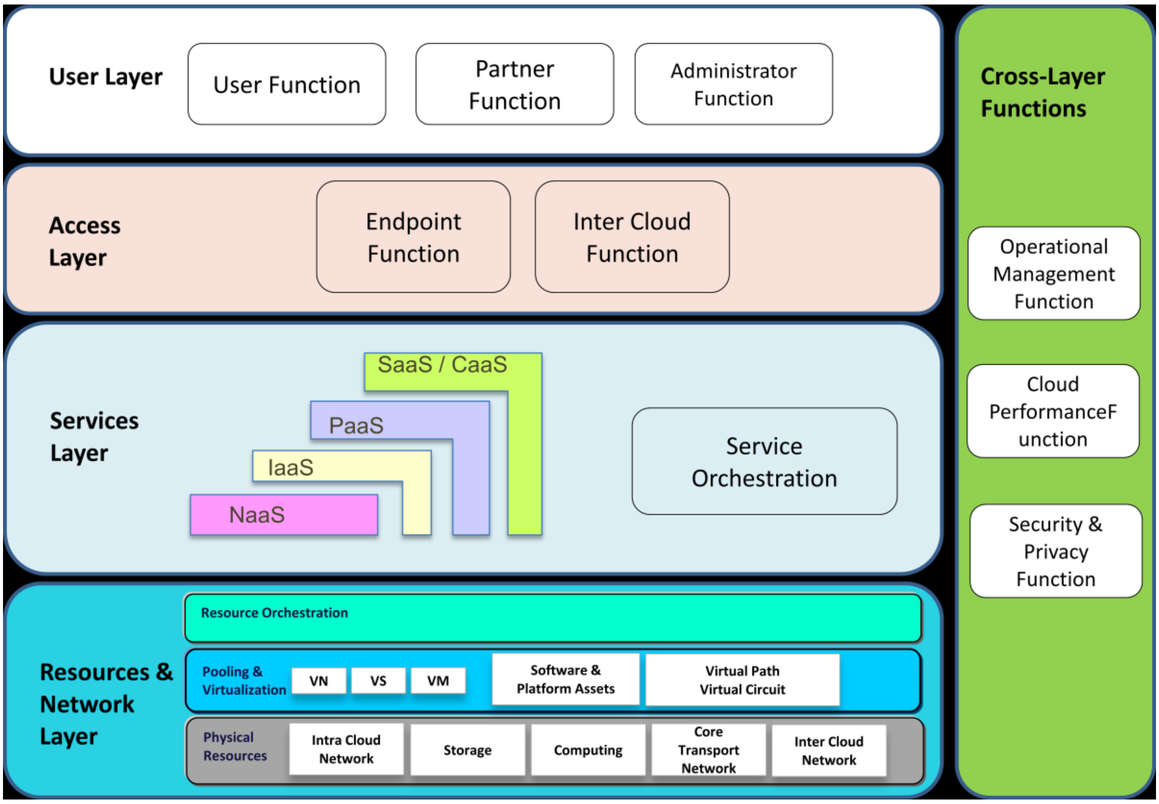

Figure 2 - Cloud computing functional reference architecture

5.1 User layer

The user layer includes the end-user function, the partner function and the administrator function.

5.1.1 User function

The user function supports a CSU to access and consume cloud services

5.1.2 Partner function

The partner function enables the CSN relationship with CSP

5.1.3 Administrator function

The administrator function supports the enterprise administrator to manage and administrate cloud resources and services within business processes

5.2 Access layer

The access layer includes the endpoint function, and the inter-cloud function.

5.2.1 Endpoint function

The endpoint function controls cloud traffic and improves cloud service delivery. This function, which has some similarity with the NGN border gateway function (located at the edge of the network), can perform traffic classification, packet marking and firewall.

This function also provides internet domain names, ports and/or Internet addresses that are "published" to the user as endpoints from where the user can (a) view and (b) request services. A published endpoint may internally "route" or "transform" the incoming service request to one or more service endpoints, based on CSP policy in effect. For instance, a published endpoint may forward the request to a location nearest to the requestor.

The endpoint function can include a subscription/publication functionality that is a subscribe-notify feature for publishing, and for obtaining notification of different cloud services and resource information from CSPs.

The endpoint function may route the requests to the inter-cloud function to be routed to external CSPs.

5.2.2 Inter-cloud function

Cloud services are expected to be offered by CSPs on a global basis. To enable delivery of such services ubiquitously, a CSP will have to establish inter-cloud connections with other CSPs who offer these services in areas that are not within the preview of the original CSP.

NOTE - Within the context of this document (functional view), the inter-cloud function is used to support the "inter-cloud" role as described in the FG Cloud Ecosystem document.

The inter-cloud function addresses different aspects of delivering any cloud service across two or more CSPs, including SLAs, resource management, performance, reliability and security; OAM&P; ordering and charging; service brokering and environmental sustainability.

The scope of the inter-connection is between two cloud domains (operated by two CSPs).

The inter-cloud function can be implemented in different manners, including inter-cloud peering, inter-cloud service broker and inter-cloud federation.

Examples of inter-cloud functional scenarios are provided in Annex V.

5.2.2.1 Inter-cloud service broker

The inter-cloud function can be implemented as inter-cloud service broker (ISB). In this case, it provides brokering service functions to CSPs or CSUs.

NOTE - The Ecosystem document [ecosystem] provides details concerning the ISB definition, scenarios and associated capability.

According to the requirements of the inter-cloud service broker (ISB) capability (as described in clause 9.4 of the Ecosystem document [ecosystem]), the inter-cloud service broker (ISB) provides and execute services of three categories:

- Service intermediation: The ISB enhances a given service by improving some specific capability and providing value-added services to CSUs. Examples of capability improvements include management of the access to cloud services, identity management, performance reporting, security enhancements, etc.
- Service aggregation: The ISB combines and integrates multiple services into one or more new services. The ISB provides data integration and ensures the secure movement of data between the CSUs and CSPs.
- Service arbitrage: Service arbitrage is similar to service aggregation except that the services being aggregated are not fixed. In this case, the ISB has the flexibility to choose services from multiple sources. For example, the ISB can use a credit-scoring service to measure and select the source with the best score.

When a CSP plays the inter-cloud service broker role, the requesting entity (CSU or CSP) continue to access the brokered services via the endpoint function. However, the implementation of the access layer will decide whether to serve the request locally, or route them to external CSPs via the inter-cloud function.

The inter-cloud function receives the requests by the endpoint function, analyses the requests, selects appropriate service logics and function patterns, executes related operations, then invokes the concrete cloud services and resources from external CSPs through cloud service adapters (by sending cloud service adaptation requests to CSPs and receiving cloud service adaptation responses from CSPs).

5.2.2.2 Inter-cloud peering

Inter-cloud peering happens when two CSPs interact with each other using already-established interfaces. Each CSP could invoke the interfaces offered by the other CSP, when needed, to perform the functions that the CSP would like to delegate to its peer CSP.

5.2.2.3 Inter-cloud federation

The inter-cloud federation capability provides an alliance among several CSPs in which mutually-trusted clouds join together logically through agreed-upon interfaces. Inter-cloud federation allows a CSP to dynamically outsource resources to other CSPs in response to demand variations.

Inter-cloud federation manages the consistency and the access controls when two or more independent CSPs share authentication, data, computing resources, command and control, or access to storage resources. Inter-cloud federation shall provide the identity syndication to support federated authentication across all sources and federated users across the different CSPs.

The inter-cloud function supports federation capability and may include federation initiator which initiates operations within a federation relationship and federation target which processes operations within a federation relationship.

5.3 Services layer

The services layer includes the service orchestration function and provides cloud services.

5.3.1 Services orchestration function

Cloud service orchestration is the process of deploying and managing “cloud services". “Cloud services” are implemented using cloud service interfaces; examples are SOAP, REST, HTML rendered pages, etc.

This function is required to provide mechanisms for:

- composing services (services provided by the CSP as well as services from other CSPs) to create new composite services;
- executing composite services.

NOTE - Composing services is the process of generation of the service logic (i.e. the logic of the service to be provided by the CSP). Executing composite services includes parsing and running the service logic.

This function also provides message routing and message exchange mechanisms within the cloud architecture and its different components as well as with external components. Message routing can be based on various criteria, e.g. context, policies. The message exchange mechanisms control the message flows between CSUs, CSP (internally) and externally with other CSPs via the inter-cloud function.

5.3.2 Cloud services

The services layer provides instances of IaaS, PaaS, SaaS, CaaS, NaaS service categories and any composition of these services.

Examples of such services are:

- The ability for users to request, configure, and execute one or more VMs
- The ability for users to invoke a software application from within a web browser
- The ability for developers to invoke programming interfaces through a cloud interface

5.4 Resources and network layer

Note: The detailed requirements and framework of the resources and network layer in this figure is defined in Technical Report *Requirements and framework architecture of cloud infrastructure*.

5.4.1 Resource orchestration

Resource orchestration is defined as the management, monitoring, and scheduling of computing, storage, and network resources into consumable services by the upper layers and users.

Resource orchestration controls the creation, modification, customization and release of virtualized resources. It holds capability directories about what is possible within a cloud segment, based upon the total resource capacity and the incremental allocations that have been done for currently honoured requests.

5.4.2 Pooling and virtualization

The pooling and virtualization of physical resources are essential means to achieve the on-demand and elastic characteristics of cloud computing. Through these processes, physical resources are turned into virtual machines, virtual storages, and virtual networks. These virtual resources are in turn managed and controlled by the resource orchestration, based on user demand. Software and platform assets in the pooling and virtualization layer are the runtime environment, applications, and other software assets used to orchestrate and implement cloud services.

"Software & platform assets" is the architectural block that represents the use of traditional software assets to implement the cloud services layer

5.4.3 Physical resources

Physical resources refer to the computing, storage, and network resources that are fundamental to providing cloud services. These resources may include those that reside inside cloud-data centres (e.g., computing servers, storage servers, and intra-cloud networks), and those that reside outside of data centres, typically networking resources, such as inter-cloud networks and core transport networks.

5.5 Cross-layer functions

Note: The detailed requirements and framework of relevant cross-layer functions (e.g., resource management and power management) in this figure are defined in the Technical Report *Requirements and framework architecture of cloud infrastructure*.

The cross-layer functions include security and privacy, cloud operational management, and cloud performance functions.

This layer caters to a CSP's need for monitoring of resources, and generates a consolidated view of current resource allocations and how efficiently the resources are being utilized. Resource monitoring allows the CSP to exercise load balance to ensure the performance of cloud services. It also flags network and service related problems such as hardware or software resource failures, missing SLA targets, or if a CSP's network is experiencing security violations or other forms of compromised situation.

As the point that collects availability, performance and security information, it is the central source of information on a CSP's service excellence.

5.5.1 Security and privacy

This section describes a high-level summary of security and privacy functions with regard to the reference architecture. For further details, please refer to the security output document.

Cloud security and privacy refer to the cross-layer functions responsible for applying security related controls to mitigate the potential threats in cloud computing environments. CSPs (CSPs) need to consider applicable privacy requirements and regulations.

Security threats can be dived into two areas, threats for CSUs and threats for CSPs.

Threats for CSUs

- Responsibility ambiguity
- Loss of governance
- Loss of trust
- CSP lock-in
- Insecure customer access
- Lack of information/asset management
- Data loss and leakage

Threats for CSPs

- Responsibility ambiguity
- Protection inconsistency
- Evolutional risks
- Business discontinuity
- Supplier lock-in
- License risks
- Bylaw conflict
- Bad integration
- Insecure administration API
- Shared environment
- Hypervisor isolation failure
- Service unavailability
- Data unreliability
- Abuse right of CSP

(These threats are more fully described in the cloud security output document)

Cloud security and privacy use a set of requirements to mitigate the above listed threats. These requirements cover actions and technologies that are deployed for both CSUs and CSPs. (Details of the requirements can be found in the cloud security output document)

Security and privacy tie in closely with SLA and service monitoring, which often depend on the user identity. The identity information must be propagated in the cloud computing environment, to enable linking of SLA and services with identity. For instance, to regulate the consumed network bandwidth to the total bandwidth the user has requested for, the monitoring point must associate the user's IP address with the identity that comes from user authentication.

5.5.1.1 VPN and firewalls

Current enterprise security comprises of two levels: (a) VPN and firewalls at the enterprise edge to restrict unauthorized users from gaining access, (b) host/service specific authentication through HTTP, SSH etc. Both forms of security need access to authentication databases.

In the cloud, the VPN/firewall security must be provided by the CSP while the host/service specific authentication is controlled by the enterprise. This is needed to enable a separation between the proprietary user information that resides in the two domains.

5.5.1.2 Application authentication is separate from VPN authentication

The CSU may have to further authenticate itself with the enterprise-user authentication mechanisms, using SSH, HTTP or other procedures. This is consistent with the “two factor authentication” widely used in all enterprises.

In case of private or federated methods of authentication, parts of the user database in the enterprise network may be used for both VPN authentication and service authentication. In the public authentication method, the CSP’s authentication may differ from the service-specific authentication configured by the cloud user.

5.5.1.3 Propagating identity information in the cloud computing environment

For several cloud services, it is important to know the user’s identity and enterprise affiliation in the cloud. For instance, to satisfy QoS requirements (bandwidth, delay) of a specific enterprise customer, the cloud needs to associate the user’s IP address with their enterprise identity. Since the user has already been authenticated at the enterprise or CSP edge, this identity needs to be propagated to cloud entities that will enable this service.

5.5.2 Cloud performance function

The cloud performance function serves the following purposes:

- To aggregate or summarize cloud network information for the management node
- To make service routing decisions by the published endpoints
- To monitor customer SLAs, and flag SLA violations
- To serve as point of monitoring and control for the management node

5.5.3 Cloud auditing

The cloud auditing function collects audit events, logging and reporting information on a per-tenant and application basis.

CSNs playing the role of auditor, as required by CSUs, can use the cloud auditing function to assure CSP’s service compliance to governmental regulatory requirements, and SLAs contracted with tenants or other tenants’ requirements.

5.5.4 Cloud operational management function

5.5.4.1 Power management

Power management is to be seen as a part of the management function, and as such it is a cross-layer function.

Power management represents a collection of IT processes and supporting technologies geared toward optimizing data-centre performance against cost and structural constraints. This includes increasing the deployable number of servers per rack when racks are subject to power or thermal limitations, and making power consumption more predictable and easier to plan for.

For more details on power management function, please refer to the infrastructure document.

5.5.4.2 Configuration function

This function is comprised of databases and provisioning information, using those cloud services that can be customized. It will include user databases for authentication and authorization, policy servers for defining end-user SLAs, geographical or legal customizations on services, name servers for describing virtual and physical hosts along with cloud domain names, automatic configuration servers for devices that can be auto configured when they are powered on, and TFTP- or HTTP-based software servers from where prior loaded images can be downloaded.

Cloud-configuration databases comprised of configuration information that is necessary to coordinate cloud services.

- **Cloud name server (CNS)**

The CNS contains DNS mappings between customer domain names and CSP domain names. These mappings help users to locate the services to their IP addresses. The inter-cloud function must make the DNS mappings after creating a service, and remove the mappings after deleting a service.

- **Cloud user database (CUD)**

The CUD is used for authenticating users against their permissions. The database is used in authenticating API requests and VPN tunnels. Details on the CUD are discussed in the security architecture proposal. The CUD will also store charge plan for users. The charge plan shall be used by the cloud-charging function to generate billing records.

- **Cloud-policy server**

Cloud service orchestration is automated and, to achieve the automation, customers must specify the policies using those services that can be orchestrated. The policy database can include specific resource policies (computer, network, storage, and firewall), and can include broader service-level policies (e.g. security, mobility, availability, and reliability). There can also be geographical, legal, SLA, and corporate policies on resources. The policy server will store the charge plans. The cloud charging function will use these policies to create billing records.

- **Cloud automatic configuration server (CACS)**

Automatic configuration of resources is needed to bring them to the point of being managed by the cloud orchestration. This can include the installation of OS/hypervisors on newly-procured servers, of network images on the network, and of security or storage devices.

Some existing methods of such early stage installations include DHCP, BOOTP, PXE, etc. Other methods might be devised for specialized cloud needs. The CACS is a collection of any, and all, of such methods to be used for automatic configuration.

- **Cloud software server (CSS)**

The CSS will host software images to be installed. These can include OS/VM images, middleware or platforms and applications. Typically, such images should be bundled along with configuration information (such as total memory, partitions needed, directories that must exist, etc.).

5.5.4.3 Charging

Charging is critical for an operational cloud-service platform. Detailed logs of the users' resource consumption should be correctly collected and retained. The charging records should carry comprehensive information about how customers use the infrastructure resources. Specifically, the resource management system should collect the following charging data:

- Computation consumption usage. This should reflect how much computation resources have been used by the customers. Data of computation consumption include the number of CPUs and the corresponding using time.
- Storage consumption usage. This should show how much storage resources have been used by the customers.
- Network bandwidth usage. This should keep the information of how much bandwidth has been used by the customers.
- Other information that may affect charging, such as delivered QoS.
- Other information for statistics.

Annex I
Service architecture for desktop as a service (DaaS)

I.1 Main functions of the DaaS service architecture

The main functions in the DaaS service architecture are given as follows:

- Connection brokering (CB) function

  A connection broker is a software program that allows a CSU to connect to an available virtualized desktop. The connection broker performs tasks which include: user authentication and license verification to validate the user and user's software; management of a virtual machine (VM) for user assignment; server monitoring to measure the activity level of a given VM, and protocol coordination to coordinate on the protocol to be used between user and server. The connection broker can also support the connection between a backup storage and virtualized desktop servers.

- Resource pooling function

  In order to provide on-demand virtualized desktop services, a resource pool can manage three different types of high-capacity software resources, such as operating systems (OS), applications and user profiles. These software resources are transferred to a certain VM in streaming form, timely, and run on the VM. A resource pool can offer provisioning information regarding the software resources on request by a CB.

  NOTE - A user profile contains individual information about hardware configuration (i.e. CPU, RAM, I/O), used OS, selected applications, and user's computing environment information (i.e. display resolution and Internet access way).

- VM infrastructure function

  The main role of the VM infrastructure is to support hardware resources and create VMs. In a virtualized desktop server, a virtualization function, called a hypervisor, is highly necessary to employ hardware resources efficiently. A hypervisor can abstract physical hardware resources and assign them dynamically to a higher level of software. Consequently, the VM infrastructure provides VMs - the "virtualized desktops" - on which user's software is operated. It is recommended that the VM infrastructure consists of a cluster environment with high availability features, and within which many running VM instances are created from the same VM template with pre-defined configuration parameters.

- Virtual desktop delivery function

  This function involves encapsulating and delivering either access to an entire information system environment, or the environment itself to a remote client device through the network. The virtual desktop delivery protocol (VDDP) is the core component to realize this function and provides the communication channels between user terminals and servers for DaaS in order to transfer all the interaction information. This information includes display information, control and configuration information, monitoring information etc.

I.2 Interaction process between client and functional components of the DaaS

Figure I.1 shows a conceptual diagram showing the interactions between client and functional components in the DaaS service architecture.

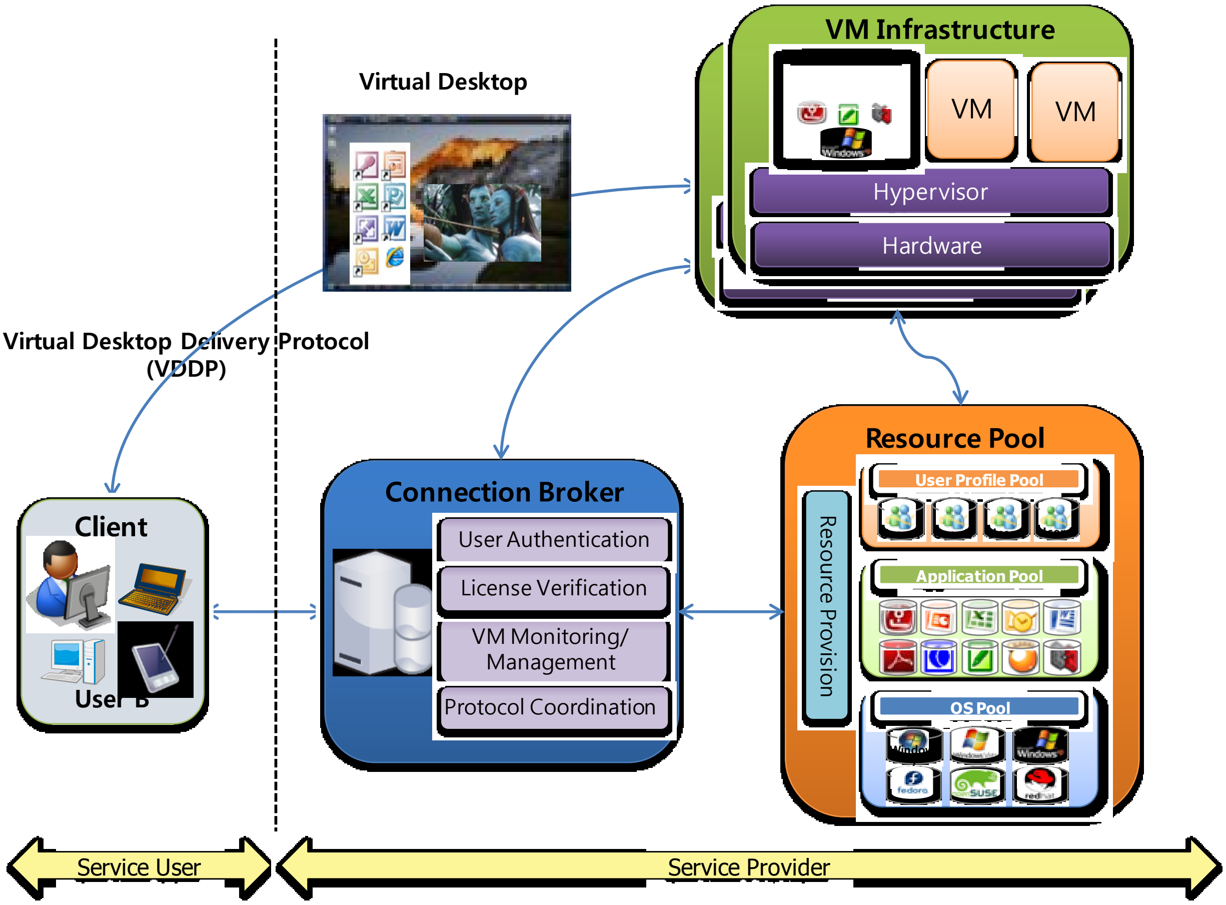

Figure I.1 - Conceptual diagram of the DaaS service architecture

The various steps of the interaction process for DaaS are as follows:

- A user accesses a CB through a security protocol (such as SSH or TLS) and the CB validates the user with a user-ID and associated password.
- The CB identifies a corresponding user profile and, in order to assign a VM, a provisioning function helps the CB to search the VM that satisfies the user's hardware configuration and is optimal to the computing environment.
- If there is no proper VM, the CB requests the VM infrastructure to create a VM by sending information of hardware configuration.

- After a VM is assigned or created, the CB applies the user profile to the VM, including installation of OS and applications, to construct a virtualized desktop.
- A connection to deliver a corresponding virtualized desktop is created in the VM infrastructure and the information of the connection is dispatched to the CB.
- The CB sends the connection information to the user and the user connects to the desktop in the VM infrastructure.
- The user communicates with the virtualized desktop through the network using the virtualized desktop delivery protocol (VDDP).
- The user executes a log-off operation to prevent the elimination of user data when he or she terminates a virtualized desktop service.
- During the log-off operation, the CB updates the modified user profile in a user profile pool to keep the most recent information, and returns the VM whose state becomes available.

Annex II
SDPaaS

SDPaaS - as defined in the ecosystem document [b-*ecosystem*] - is the capability provided to the CSU to use service delivery platform (SDP) functionalities, and services provided by a CSP, and the capability provided to a CSP to deploy, control and manage service delivery platform functionalities.

SDPaaS allows a CSU to access, as a service, functionalities and services offered by a CSP similar to those offered by a traditional SDP. Services offered by a traditional SDP include different types of services (e.g. telecom services, Internet access and portal services, etc.).

Functionalities offered by a traditional SDP include service creation (functional group), service execution (functional group) and service delivery management (functional group) [b-Y.2240]:

- The service creation functional group provides capabilities to realize an application development environment for application developers.
- The service execution functional group provides capabilities to support a service execution environment.
- The service delivery management functional group provides capabilities to realize the management of different aspects, provisioning of applications and charging, to ensure the proper functioning of the service creation and service execution functional groups, and to provide associated delivery functionalities.

SDPaaS may be implemented via the utilization and intermediation of different SaaS/CaaS and PaaS cloud services, located in the cloud services layer of the cloud reference architecture.

SDPaaS exposes services and functionalities, similar to those offered by a traditional SDP, as cloud services:

- Services are provided as SaaS/CaaS services.
- Functionalities are provided as PaaS services (NOTE 1).

NOTE 1 - SDPaaS enables a CSP to deploy, control and manage SDP functionalities. The service orchestration function residing in the cloud services layer may provide support in that respect, including for integration and intermediation of services and functionalities with other cloud services to build more complex or custom-purpose services. However, details about SDPaaS with respect to the functions of the cloud computing reference architecture are out of the scope of this document**.**

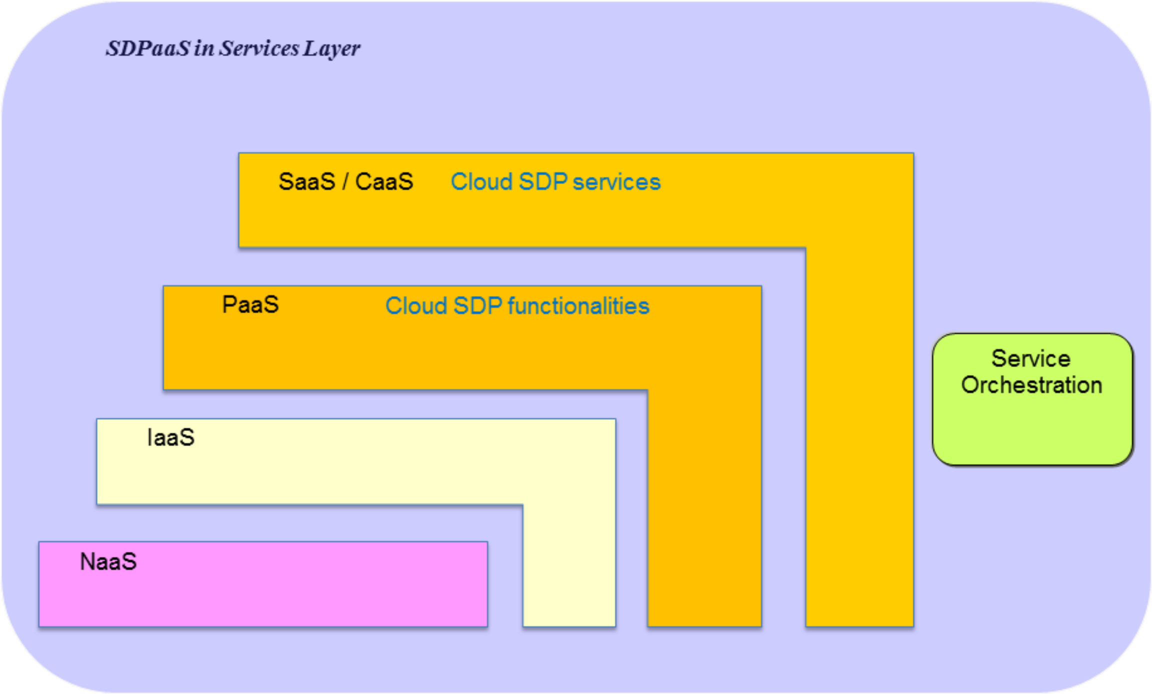

Figure II.1 - SDPaaS in the services layer of the cloud reference architecture

Annex III
The role of business processes

By *business processes* we mean the following kinds of tasks:

- Cloud-service lifecycle management
- Definition of cloud services and their associated SLAs
- Policies on resource allocation (e.g. restrict a user's disk usage to N GB, dynamically increase number of servers for service X to a max of five servers, etc.)
- Modelling and simulation – capacity growth planning (e.g. what will the CPU utilization on the application server be when the number of users is doubled?)
- Provisioning and pre-provisioning resources based on forecasts
- Periodic tasks (e.g. taking backups, generating reports)
- Monitoring service availability and actions to be taken on service failures

Cloud services need to be managed and monitored by business processes. These processes can include SLAs by which the service is measured, in addition to normal IT processes such as provisioning, monitoring and modifying cloud services.

A particular customer's business process might, for instance, define procedures on recovery actions to be taken when a particular cloud network entity fails. This can include taking corrective actions (such as re-provision or migration of failed resources), notifying concerned authorities of the failure, and others. The processes might also include taking automated backups, regular monitoring of certain cloud services, synchronizing user-databases between CSPs and customers, etc.

The business processes might be unique to a given customer; however, the building blocks of these processes will be common. Typically, these processes will in turn utilize the primitives exposed by cloud APIs. The business block must therefore sit "on top of" the cloud API block.

The business process function sits on top of the endpoint function, which contains the end-user control and API. The API exposes primitives to perform atomic actions such as create, delete, modify, and configure resources. The business processes abstract the functionality in the cloud API into lifecycle management procedures, policies, and SLAs, which a customer wants to apply to cloud services. Business processes can, for example, be abstractions of procedures and policies of a company's IT department.

The end-user control and API contain the cloud access API for requesting and customizing services, security procedures for user clients accessing cloud services, and provide the manageability interfaces for monitoring. This functionality is part of the access layer (see Figure 2).

Service orchestration basically cares for policy driven automation of resource creation, allocation, tearing and operational optimization through resource mobility (located in the service orchestration function), network monitoring, SLA measurement and compliance (located in cross-layer monitoring and SLA) – see Figure 2.

Virtualized resource management advertises available resources for allocation, as well as virtualization of resources for multi-tenant access. For creation, customization and tearing of resources, and virtualization formats can be used (like OVF). This functionality is part of the resources and network layer (see Figure 2).

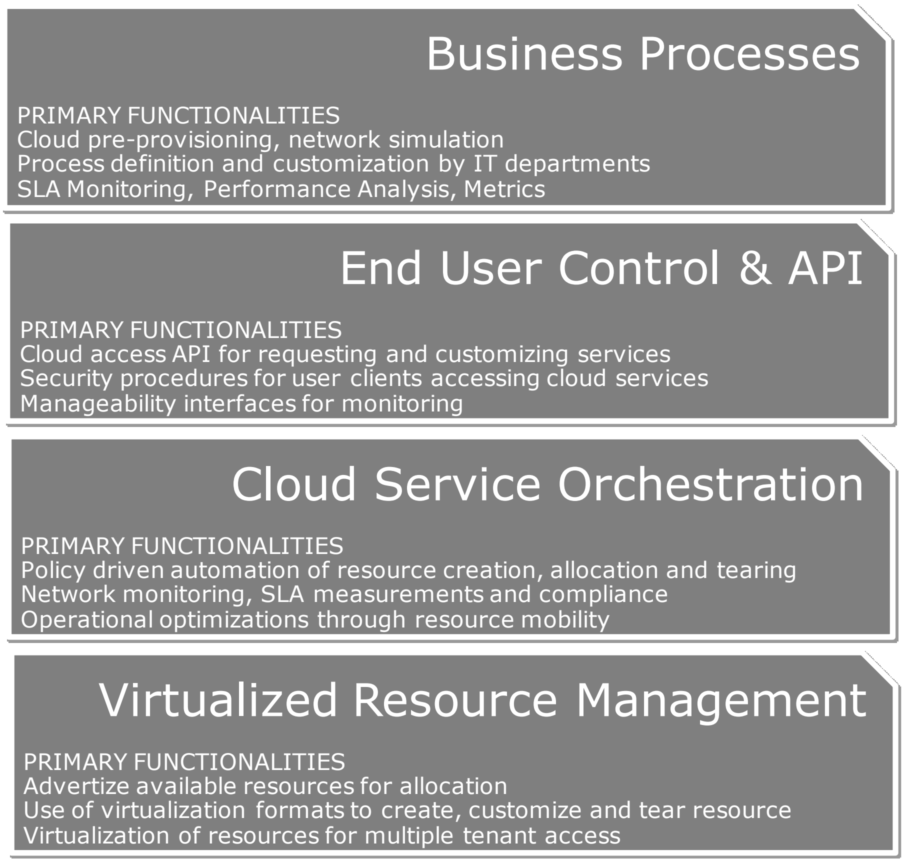
Business Processes
PRIMARY FUNCTIONALITIES
Cloud pre-provisioning, network simulation
Process definition and customization by IT departments
SLA Monitoring, Performance Analysis, Metrics
End User Control & API
PRIMARY FUNCTIONALITIES
Cloud access API for requesting and customizing services
Security procedures for user clients accessing cloud services
Manageability interfaces for monitoring
Cloud Service Orchestration
PRIMARY FUNCTIONALITIES
Policy driven automation of resource creation, allocation and tearing
Network monitoring, SLA measurements and compliance
Operational optimizations through resource mobility
Virtualized Resource Management
PRIMARY FUNCTIONALITIES
Advertize available resources for allocation
Use of virtualization formats to create, customize and tear resource
Virtualization of resources for multiple tenant access

Figure III.1 – The role of business processes

The business processes must eventually be described in some language. Examples of such languages and procedures include BPEL (Business Process Execution Language) and BPMN (Business Process Modelling Notation). BPEL is an XML based “orchestration language”. BPMN is a graphical language for representations of business processes.

Annex IV
Example of modelling real-world cloud service using ITU-T FGCC RA

Internet television ecosystem

This example shows a federated ecosystem of cloud-based entertainment services, where multiple commercial retailers offer interoperable streaming and download of premium video content, ownership of which is recorded by a central rights locker.

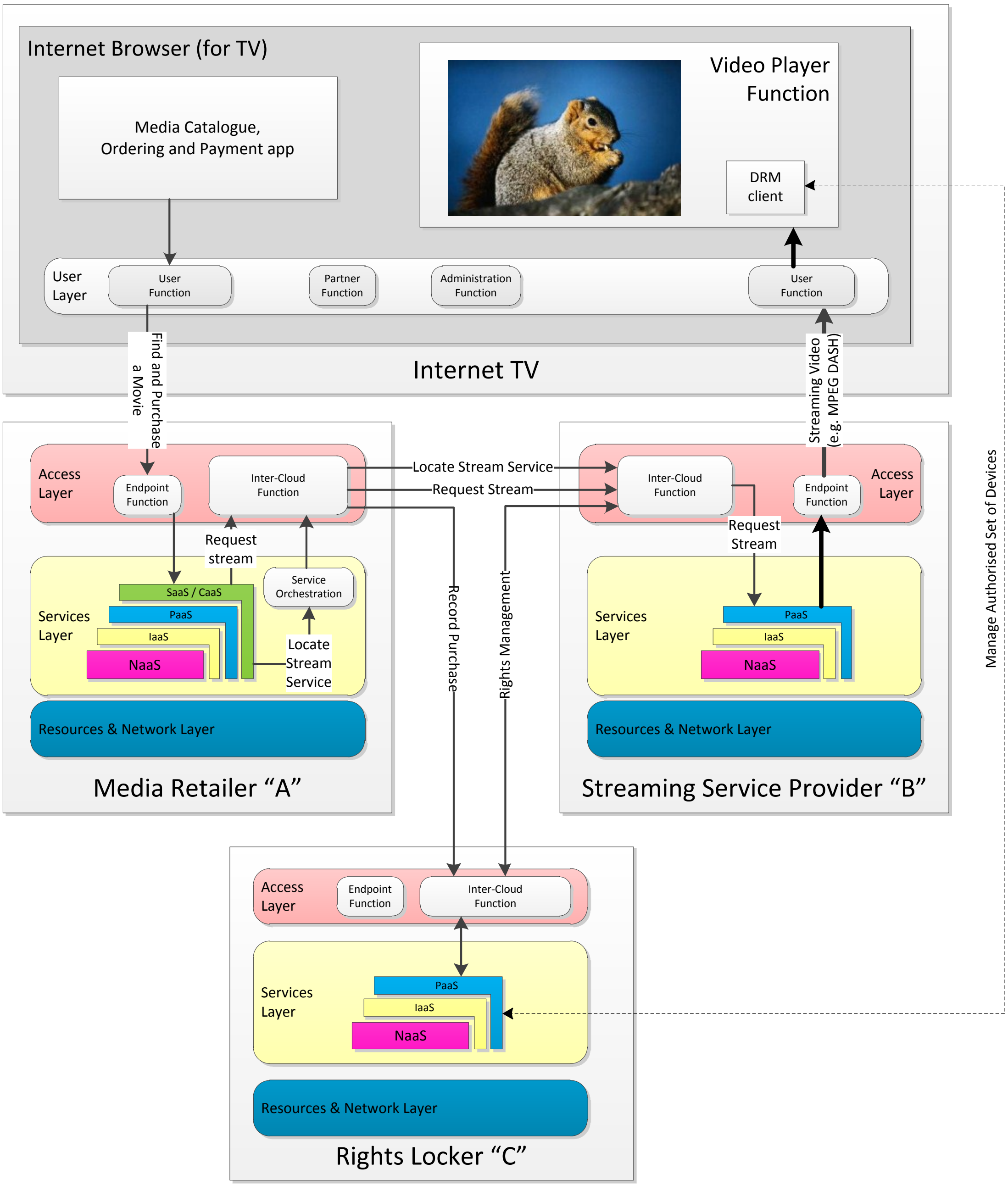

Figure IV.1 - Reference model applied to a cloud entertainment service

Within this scenario, the following participants are represented:

Entity	Description
Internet TV (CSU end-user)	A consumer-oriented client device that offers access to audio-visual entertainment media. In this scenario, the device includes an HTML5 browser which is able to execute CSU applications such as web pages and apps, and which supports embedded video/media components natively in the page. Note: The player functionality of the browser also incorporates DRM functions to enable use of protected premium content. The DRM client will read any licence or DRM header information from the received stream, and use this to obtain the necessary content keys from the appropriate DRM server. This process is always proprietary, so is not shown in this diagram. The actual DRM server may be operated by the Media Retailer "A", the Video Streaming Service CSP "B", a content CSP (e.g. studio or PayTV channel, not shown) or by another entity entirely, but this is invisible to the consumer.
Media Retailer "A" (CSP)	Company "A" operates a cloud service that offers a consumer-facing catalogue of entertainment content, with recommendations, payment processing, etc. Such a retailer could be a video-retailing website, a telco IPTV service, a hotel entertainment system, or an adjunct to a Pay-TV or other broadcaster, or similar. "A" is able to offer content for sale or rent to the consumer through a cloud application, collect payment, and authorise use of the media. "A" is also able to determine that the consumer already owns a given title by reference to the Rights Locker cloud service operated by company "C". Company "A" can use cloud service orchestration to find a suitable cloud streaming or download source for a given media delivery. In this scenario, "A" is responsible for finding and invoking a suitable Streaming Service CSP (company "B") from the cloud, and will base this decision on location, content availability, available stream capacity, cost to the retailer, etc. Once this is determined, company "A"'s application on the Internet TV will pass the appropriate URL to the media player function (by pointing to an MPEG DASH MPD file). "A"'s Media Retailer service is implemented as a SaaS since it communicates directly with the human user through a cloud application running in the browser and back-end, so as to provide a rich catalogue experience with recommendations, ratings, social media connections, account management, billing, special offers, etc.
Streaming Service CSP "B" (CSP)	In this scenario, "B" provides a service that fulfils the delivery of media to consumers when requested by another cloud service. "B" provides stream fulfilment for numerous Media Retailers in addition to "A". "B" also delivers DRM licences for any protected content, when authorised to do so by the relevant Media Retailer. "B"'s streaming service is implemented as a PaaS, since it is a generic streaming platform that delivers stored content using standardised streaming protocols. Differentiation is based on "B"'s location, catalogue, and pricing of the service. In this scenario, "B" will bill "A" for the cost of media delivery.

Entity	Description
Rights Locker "C" (CSP)	Company "C" acts as the ecosystem's Rights Locker, the central governing entity within the ecosystem. "C" keeps a record of which devices are authorised and which media titles are owned by a given consumer account. In this way, content that was bought through one Media Retailer may also be watched using the service of another Media Retailer, possibly using the same or a different Streaming Service CSP. In this example, "C" is also responsible for authorising a set of devices to use the consumer's account. This requires tracking the various DRMs that are in use, verifying the total number of authorised devices, and supplying them with "domain" credentials for each DRM as required. In this way, content that is delivered to the consumer can be played on any of the consumer's authorised devices without fear of it being more widely distributed. "C"'s Rights Locker service is implemented as a PaaS, since for this scenario it is invoked by "A"'s application, not by the human user. While "C"'s Rights Locker service provides a rich and complex API, and is specific to this particular vertical application, it does not in itself provide a service to the consumer. The only direct consumer UI is that "C" provides a web-based account management page; however this is a "management plane" function and not part of daily consumer operation (such as buying and/or watching a movie).

Annex V
Examples of inter-cloud usage options

The following two diagrams offer two separate scenarios where a CSP act as inter-cloud service broker (ISB) using functions within different layers, showing the intended flexibility of the reference architecture.

Figure V.1 describes a scenario where a CSP is playing the inter-cloud role to subscribe/ notify from various CSPs.

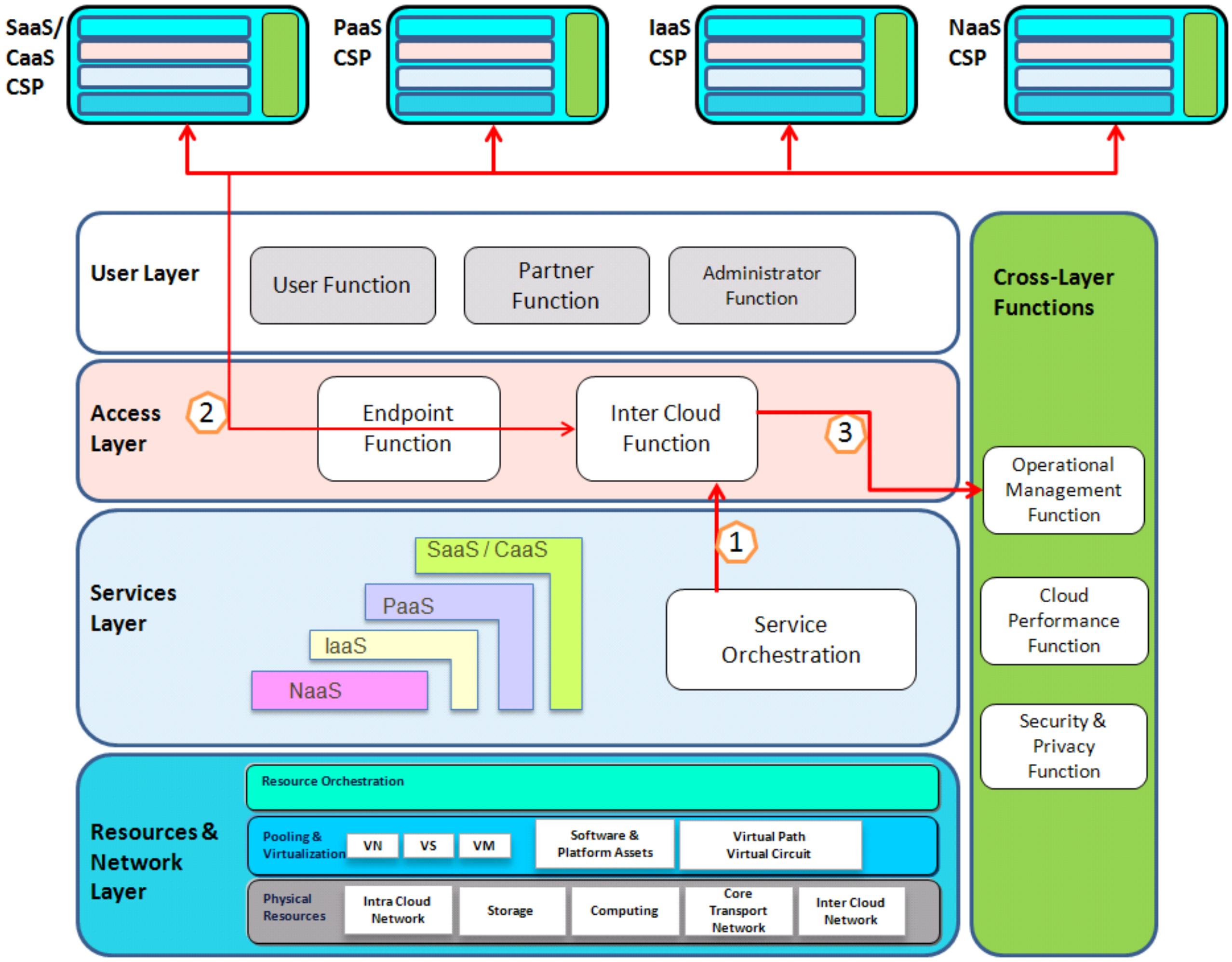

Figure V.1 – Diagram for subscription/ notification

1. A cloud workload/application, or the "Service Orchestration", inside the Services Layer determines to use external cloud resources according to CSP's policies & customer SLA requirements.
2. The "Inter-Cloud Function" in the Access Layer through the "Endpoint Function" in the Access Layer to subscribe and get notification of different cloud services and resource information from various CSPs (e.g. SaaS/CaaS CSP, PaaS CSP, IaaS CSP, NaaS CSP).

3. The CSP playing the Inter-Cloud Service Broker role through the “Inter-Cloud Function” in the Access Layer to store the notification information into the “Operational Management Function” in Cross-Layer Functions.

Figure V.2 describes a scenario where a user accesses CSP playing the inter-cloud role to get composite cloud services.

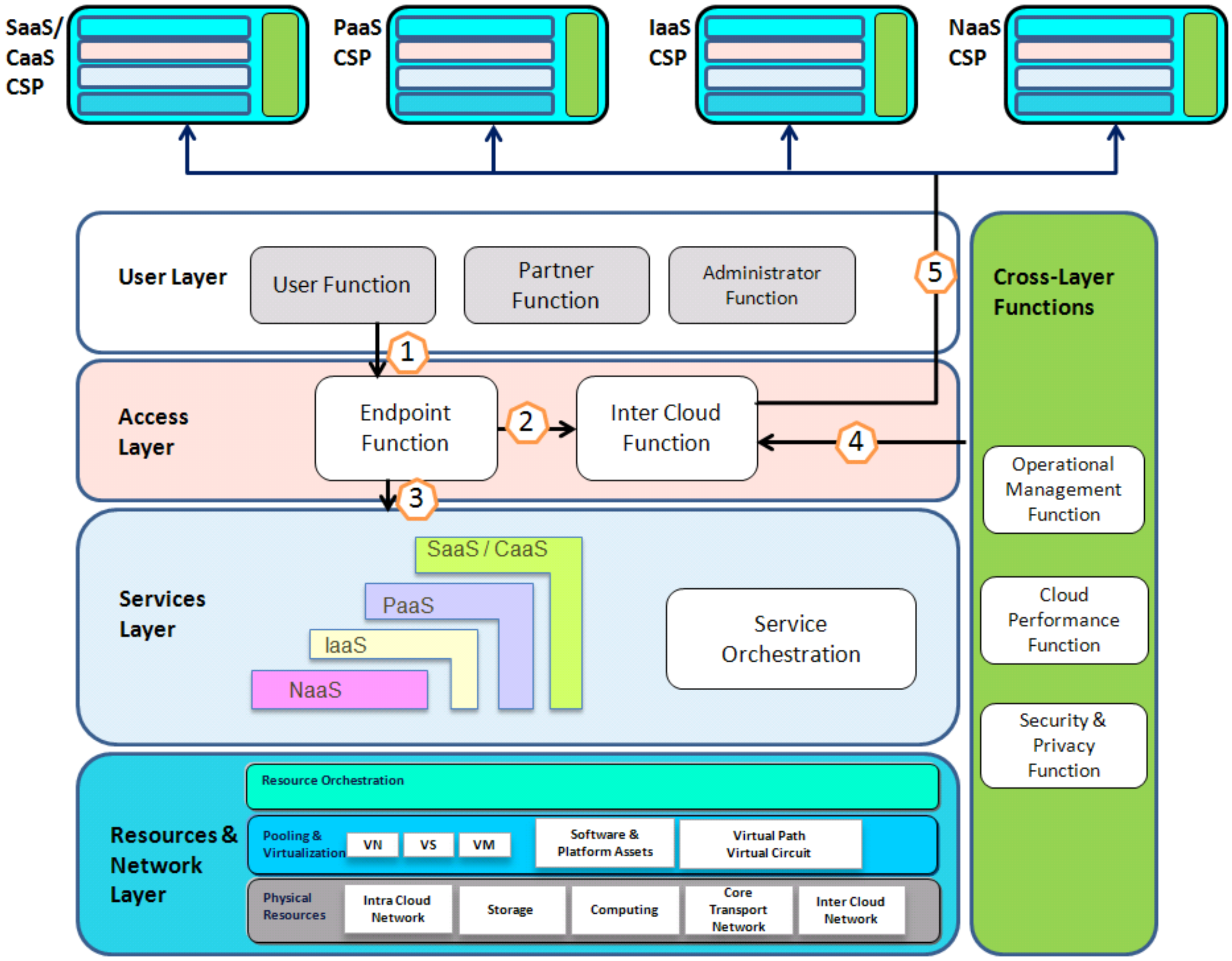

Figure V.2 – Diagram for accessing composite cloud services

1. A CSU accesses a CSP playing the Inter-Cloud role through the “User Function” in the User Layer to get composite services including brokering and local implementation cloud services. The “User Function” in the User Layer invokes the “Endpoint function” in the Access Layer.
2. The “Endpoint function” in the Access Layer routes the brokered calls to the Inter-Cloud functions in the Access Layer.
3. The CSP playing the Inter-Cloud role provides local services and resources to CSU.
4. The Inter-Cloud function in the Access Layer invokes the Cross-Layer Functions to get stored various CSPs’ services and resources information.

5. The Inter-Cloud function in the Access Layer invokes and adapts outside services and resources to CSU.

Inter-cloud federation option

Figure V.3 describes a scenario where trusted CSPs (CSPs) logically join together by integrating their resources.

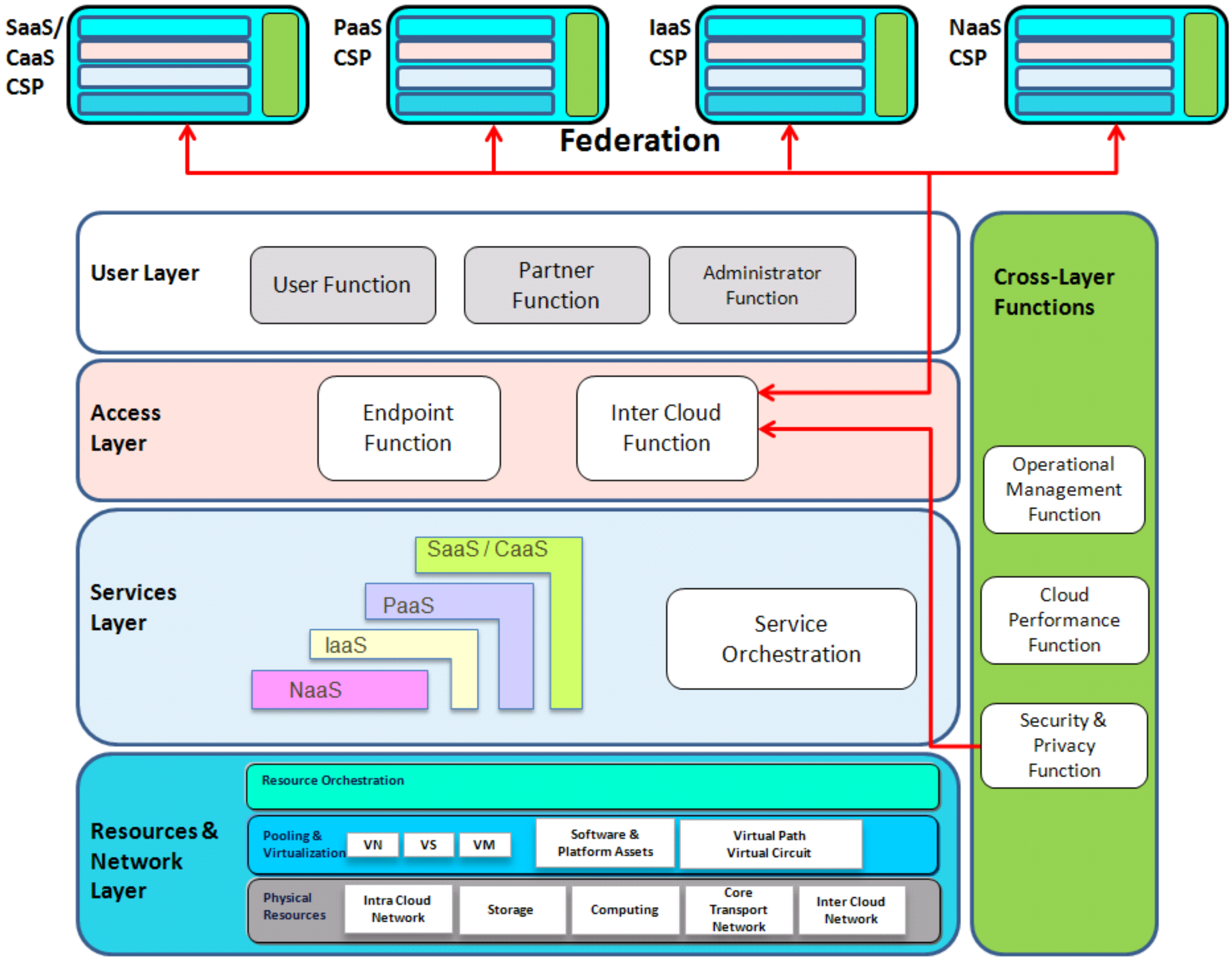

Figure V.3 – Diagram for inter-cloud federation

Inter-cloud peering usage options

Figure V.4 describes a scenario where two CSPs (CSPs) interwork directly with each other.

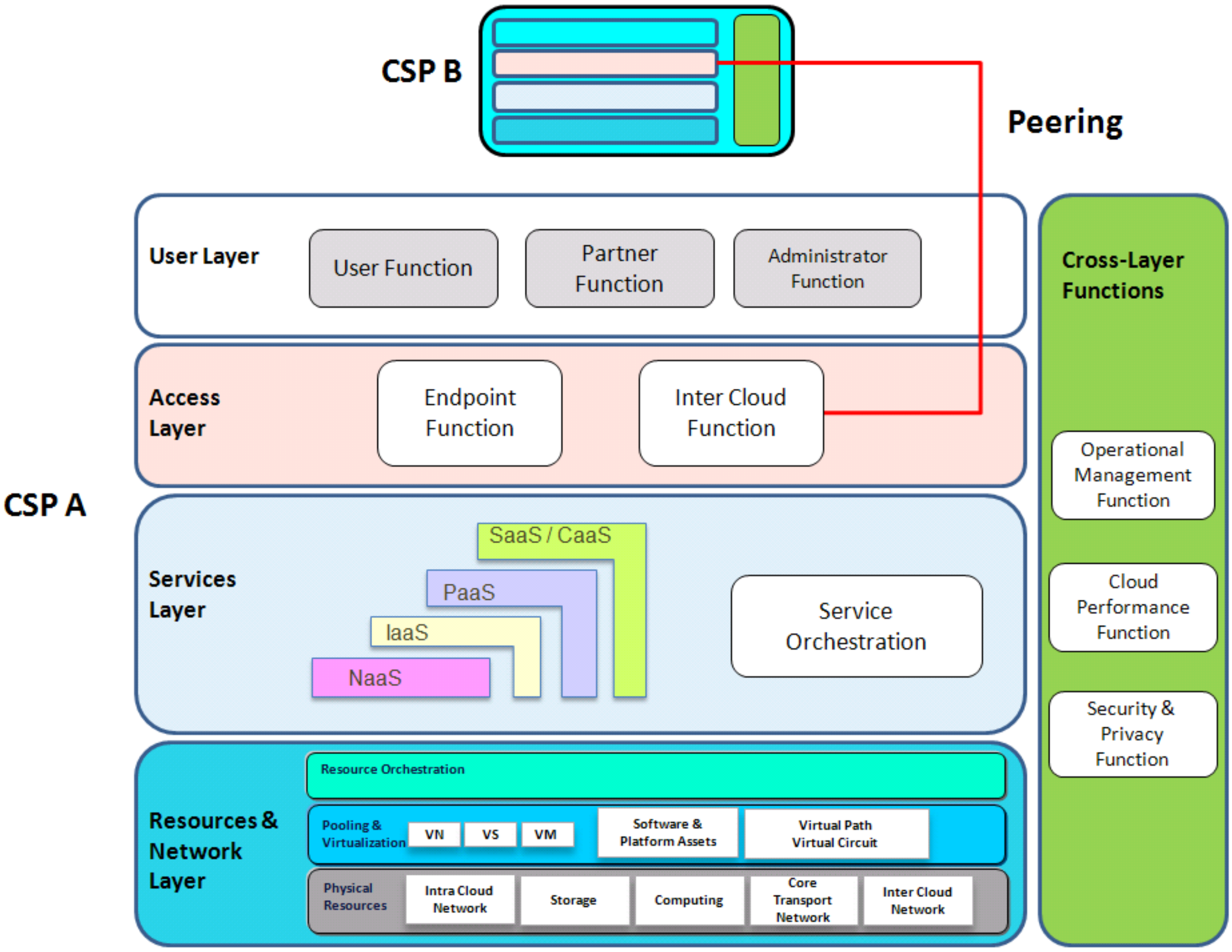

Figure V.4 – Diagram for inter-cloud peering

Annex VI
Examples of user-access options

The following two diagrams offer two separate scenarios where a cloud user accesses the cloud services through interfaces at different layers, showing the intended flexibility of the reference architecture.

Figure VI.1 describes a scenario where a user accesses the service layer through the "Endpoint Function" in the access layer:

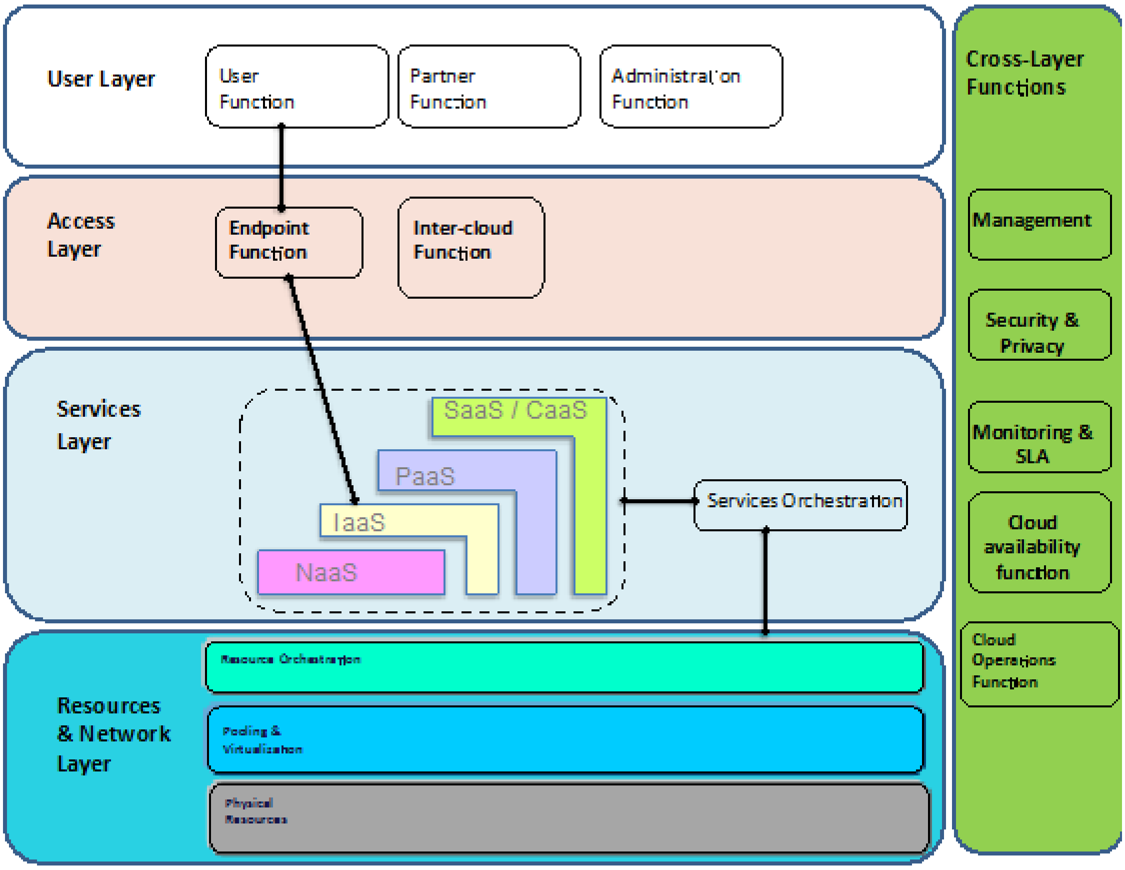

Figure VI.1 – Scenario using the access layer

The Figure VI.2 describes a scenario where a user accesses the service layer directly, bypassing the access layer:

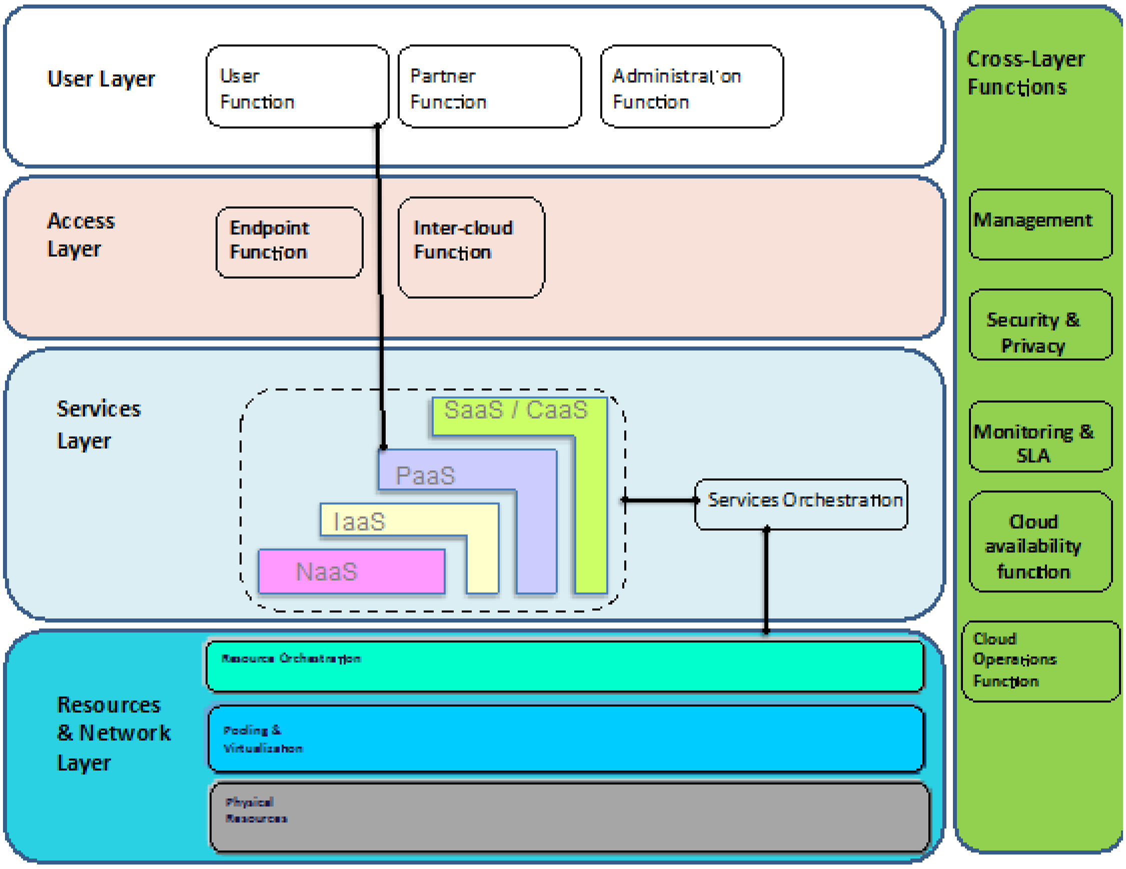

Figure VI.2 – Scenario by-passing the access layer

Bibliography

[b-ITU-T Y.2240] *Recommendation ITU-T Y.2240, Requirements and capabilities for NGN service integration and delivery environment*

[b-NIST DFN] NIST, *The NIST Definition of Cloud Computing*, version 15 (2009), http://csrc.nist.gov/groups/SNS/cloud-computing/

[b-DMTF OVF] DMTF DSP0243, *Open Virtualization Format Specification* (January 2010)

[b-DMTF CIMI] DMTF DSP0263, *Cloud Infrastructure Management Interface (CIMI) Model and REST Interface over HTTP* (Sep 2010)

[b-OGF OCCI] OGF GFD.184, *Open Cloud Computing Interface - Infrastructure* (April 2011)

[b-SNIA CDMI] SNIA *Cloud Data Management Interface*(September 2011)