

# Unit 5

## Cloud Computing Reference Architecture

### CLOUD COMPUTING REFERENCE ARCHITECTURE

Everything in the Cloud is offered in the form of services.

The service models in Cloud Computing unified the resource delivery and utilization in the form of services, or more specifically, in the form of programmable APIs, no matter it is hardware (network, server, storage) or software (applications, business processes). Therefore, service oriented management capabilities are indispensable in a Cloud Computing platform. A Cloud Computing platform should be able to support creation, delivery, provision and consumption of different service models, which makes the SOA best technique for the realization of Cloud Computing platform. On the other hand, Cloud Computing has its own characteristics. The commercial attraction of Cloud Computing comes from the advantage that it promises to offer cheap services according to the needs of its customers. Therefore, the key enabling technologies such as virtualization, on-demand service provisioning, multi-tenancy, and interoperability should be supported in the Cloud Computing

Reference Architecture (CCRA).

The reference architecture proposed in this paper, therefore, is based on the basic structure of the Open Group SOA Reference Architecture with specific extensions to support the characteristics of Cloud Computing.

The proposed reference architecture should be applicable to all four types of cloud deployment models, including public clouds, private clouds, community clouds, hybrid clouds when deployed differently.

#### A. *Objective and Principles*

The objective of this CCRA is to provide a blueprint for creating or evaluating Cloud Computing architecture. It provides templates and guidelines for public, enterprise and solution architects as well as software engineering roles within the software development life-cycle.

To facilitate the design decision and solution reuse, the concerns in the Cloud Computing design are divided into architectural layers. Requirements for each layer are identified as capabilities, and a set of ABBs are defined. ABB is the smallest constituent of the architecture model that describes a single aspect of the overall model.

Each layer contains a set of ABBs that define the key responsibilities of that layer. In addition, ABBs are connected to one another across layers and thus provide a natural definition of the association between layers. The particular connection between ABBs that recur consistently in order to solve certain classes of problems can be thought of as an architecture solution pattern.

CCRA provides a co-innovation and co-production framework for cloud vendors, cloud partners, and cloud consumers to work together. We identify guiding principles that are useful in the practice of Cloud Computing architecture design.

### **Principle 1: Virtualization Support**

The economical benefits brought by Cloud Computing come from its scaling ability of sharing the hardware and software resources among users. Virtualization is the key enabling technique. The Cloud Virtualization Management capabilities covers software image management, hardware virtualization, should be included in the design.

### **Principle 2: Service Management Support**

As stated above, the resources in Cloud Computing environment are encapsulated and used in the form of common reusable services, exposed as programmable APIs. The Service orientation service management capabilities such as Service encapsulation, composition, provision and subscription are supported in CCRA in Cloud Service Layer and Cloud Process Layer, and should be included in the design.

### **Principle 3: On-demand Service Provisioning and Subscription Support**

One of the key characteristics of Cloud Computing is that it provides service according to the needs of its customers. On-demand service provision and subscription capabilities are supported in Cloud Process Layer and should be included in the design.

### **Principle 4: Interoperability Support**

Interoperability is a necessity while a challenging problem in Cloud Computing. Seamlessly transition of application and data will require industry standards developed and followed by all those in the industry. Information representation and message exchange of Cloud Computing resources are very important to enable the collaborative and effective features of Cloud Computing. Unified information Representation and exchange capabilities should be supported[15]. On semantic level, all the mapping rules between heterogeneous service specification languages and ontology from various domains are maintained in the Cloud Interoperability Management. Semantic interoperability is supported with Interoperability Enabler by providing mapping rules between different Domain Ontology.

## *B. SOA RA and CCRA*

As the SOA RA is the base of the CCRA, in this section, we will give a brief introduction on the OpenGroup SOA RA. The SOA RA is intended as a set of best-practices and guidance on creating successful architectures using the SOA paradigm.

In SOA RA, there are nine layers, including five horizontal layers: Operational Systems, Service Components, Services, Business Processes and Consumer interfaces and four vertical layers: Governance, Information, Quality of Service (QoS) and Integration. The five horizontal layers are more functional in nature and relate to the functionality of the SOA solutions. The vertical layers are supportive of crosscutting concerns that span the functional layers but are clustered around independent notions themselves that crosscutting

concerns of the SOA architectural style.

CCRA inherits the same layered architecture of SOA RA. The five horizontal layers are very much alike to the SOA, with extension to support virtualization, on-demand service provision and selection in Cloud Computing. The four cross cutting layers in the SOA RA are also important for all cloud architectures and solutions as well. In CCRA, the four cross cutting layers, especially the layers of QoS, Governance are extended. New building blocks are added in order to support interoperability (Cloud Interoperability Management) and multi-tenancy (Cloud Ecosystem Management). The integration and information layers are still important and must be considered in the development of any Cloud architecture and solution architecture. However, Cloud does not introduce any new principles or concerns to these cross cutting layers.

Interoperability is a challenging goal faced by Cloud Computing, and we do not try to address all the interoperability issues in this proposal. To be specific, we intend to promote the semantic interoperability of Cloud services by incorporating Cloud Interoperability Management (CIM) in the Governance Layer using metadata registration with data semantics.

The concepts and architectural elements not depicted in the CCRA are still implied and presented via its SOA RA heritage.

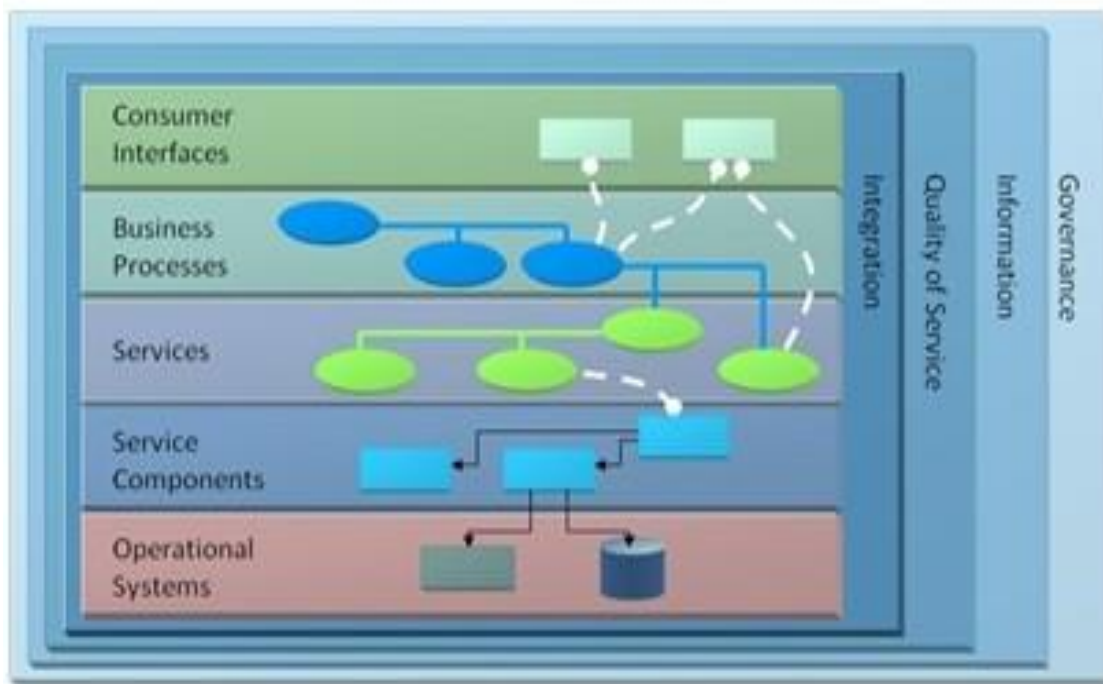


Figure 1. SOA Reference Architecture Overview[7]

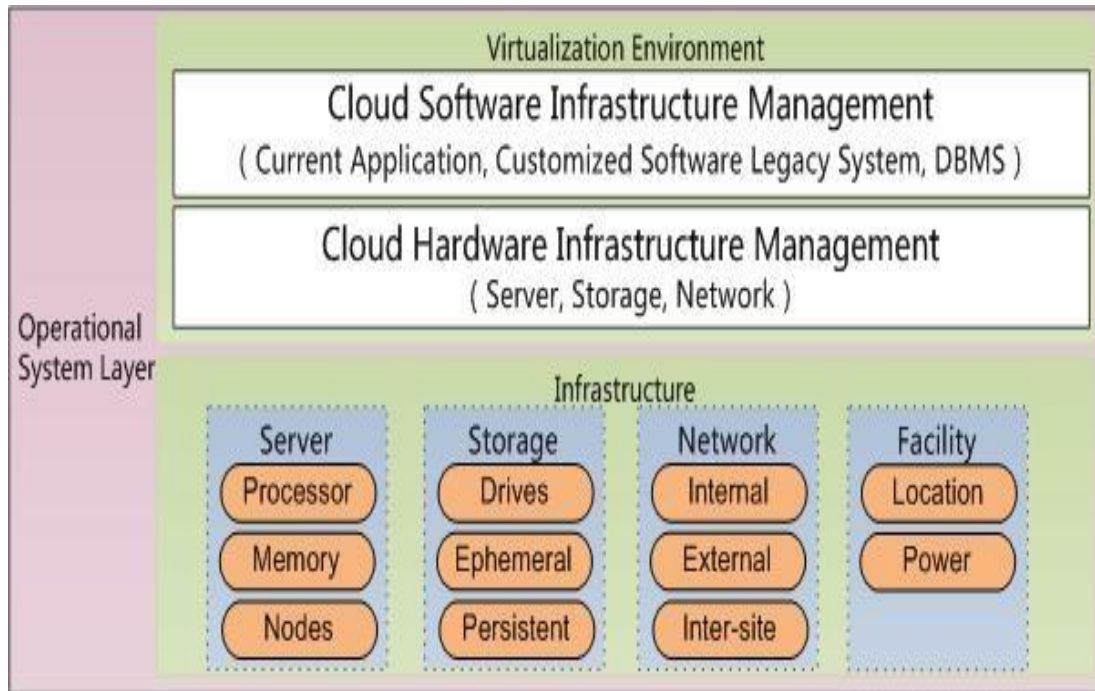


Figure 3. ABBs in Operational System Layer

#### A. Overview of the CCRA

Figure 2 depicts a Cloud Computing Reference Architecture as a set of logical layers. There are five horizontal layers that are more functional in nature and relate to the functionality of the Cloud Computing solution. The vertical layers support cross cutting concerns that span the functional layers but are clustered around independent notions themselves as cross-cutting concerns of the Cloud Computing architectural style.

#### B. Layers of the CCRA

1) *Operational System Layer*: The Operational System Layer comprises of both hardware and software support for the operation of Cloud Services. It is composed of Infrastructure and Virtualization Environment Management capabilities.

Infrastructure includes all the elements needed by the Cloud Service Vendor to provide Cloud Service, including facilities, server, storage, and network resources, how these resources are wired up, placed within a data center, etc. The infrastructure element is purely scoped to the hardware infrastructure and does not include the virtualization management functionality, which is covered by Virtualization Environment.

The Virtualization Environment Layer is used for enabling virtualization of cloud computing infrastructures in the Cloud Computing environment. According to the different types of cloud computing infrastructures, The Virtualization Environment Layer provides different capabilities: Cloud Hardware Infrastructure Management and Cloud Software

Infrastructure Management. Cloud Hardware Infrastructure Management is used to manage hardware equipments in plug-and-play model. That is to say, in this mode, hardware equipments can be added or removed without affecting the normal operations of other equipments in the system. Of course, performance or storage spaces may be dynamically changed due to those add and remove actions. The Cloud Software Infrastructure Management is used to implement software virtualization, i.e., to use software image management or software code virtualization technology to enable software sharing.

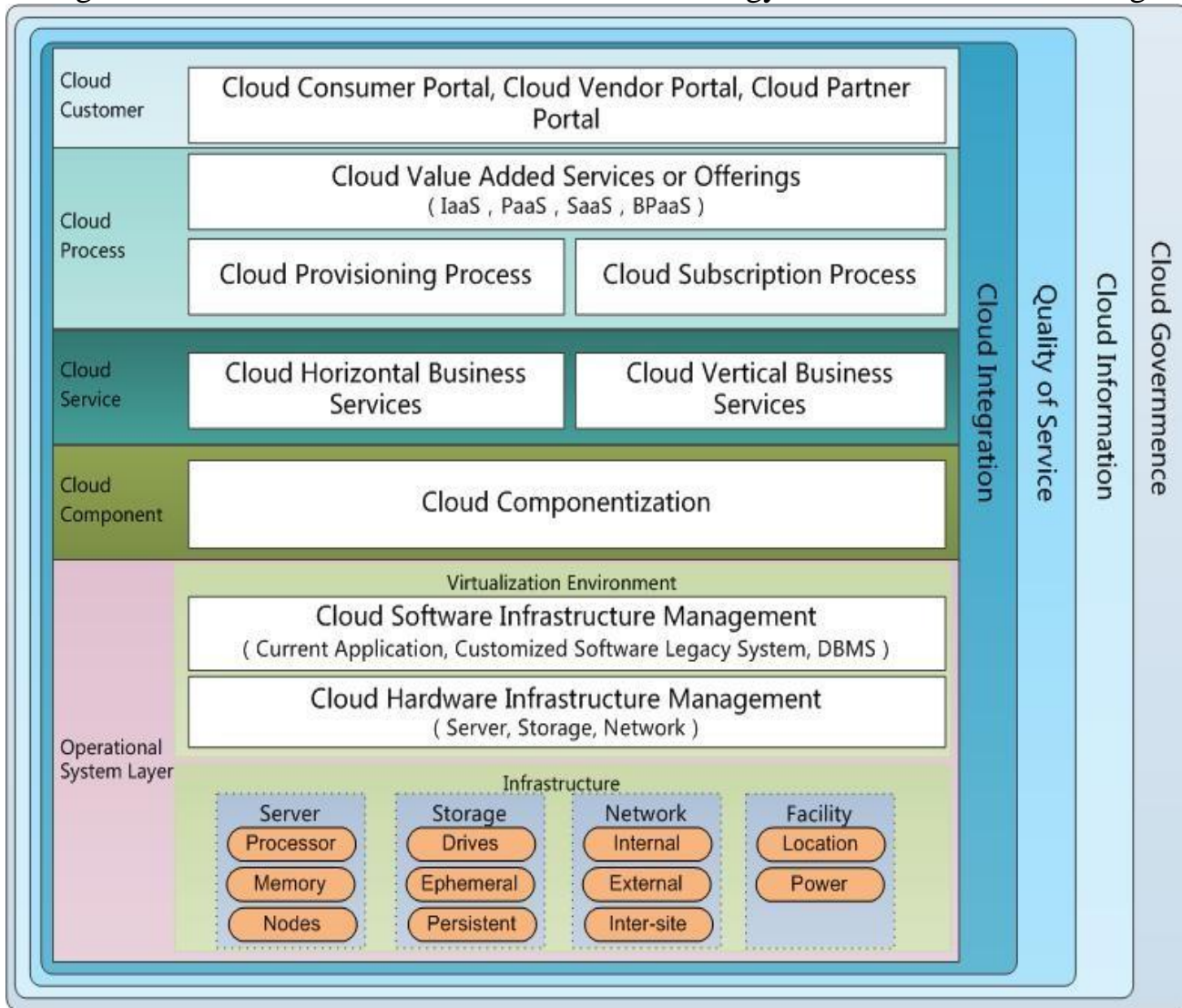


Figure 2. Cloud Computing Reference Architecture Overview

1) *Cloud Component Layer:* The Cloud Component Layer has a collection of cloud components, which could be the smallest computing units or the basic functions components. Most of the components in this layer created as results from the Virtualization Environment

Layer after the cloud hardware infrastructure management and the cloud software infrastructure management in the Virtualization Environment Layer, and some of them may be provided by cloud partner or collected from the Internet.

Cloud Component Layer is the extension of Service Component Layer in SOA RA to include virtualized hardware and software resources in the form of IT services, in addition to software components. These components provide the implementation or "realization" for cloud services and their operations.

Similarly, the layer also contains the functional and technical components that facilitate a Cloud Component to realize one or more cloud services. Cloud components reflect the definition of the service they represent, both in functionality and quality of service. They "bind" the service contract/specification to the implementation of the service in the Virtualization Environment Layer. Cloud components are hosted in containers which support the service specifications.

- 1) *Cloud Service Layer*: In addition to the virtualization characteristic, service-orientation is another driving force to enable Cloud Computing to further realize the business value from asset reusability, composite applications, and Mashup services. In the Cloud Service Layer, there are two major types of common reusable services: Cloud Horizontal and Vertical Business Services.

The Cloud Services Layer consists of all the services defined within the cloud computing reference architecture. This layer can be thought of as containing the service descriptions for business capabilities and services as well as their IT manifestation during design time, as well as service contract and descriptions that will be used at runtime. There are a few capabilities that Cloud Service Layer needs to support, including service definition, On Demand Service Discovery, Service Runtime Enablement and Service Clustering.

- 2) *Cloud Process Layer*: The Cloud Process Layer is used to provide capabilities for cloud service provisioning and offering. Capabilities for cloud service provisioning include cloud provisioning process and cloud subscription process. Cloud vendors and partners cloud use cloud -



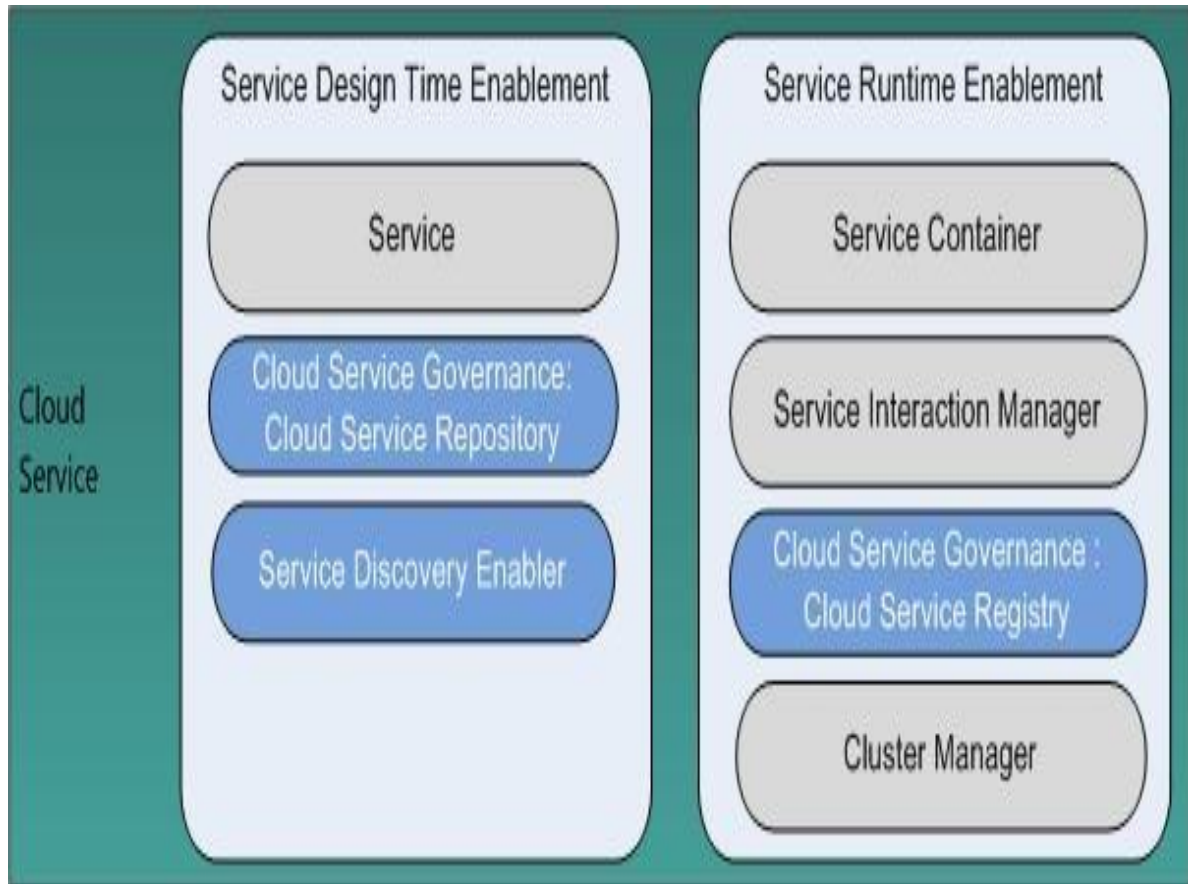


Figure 4. ABBs in the Cloud Service Layer

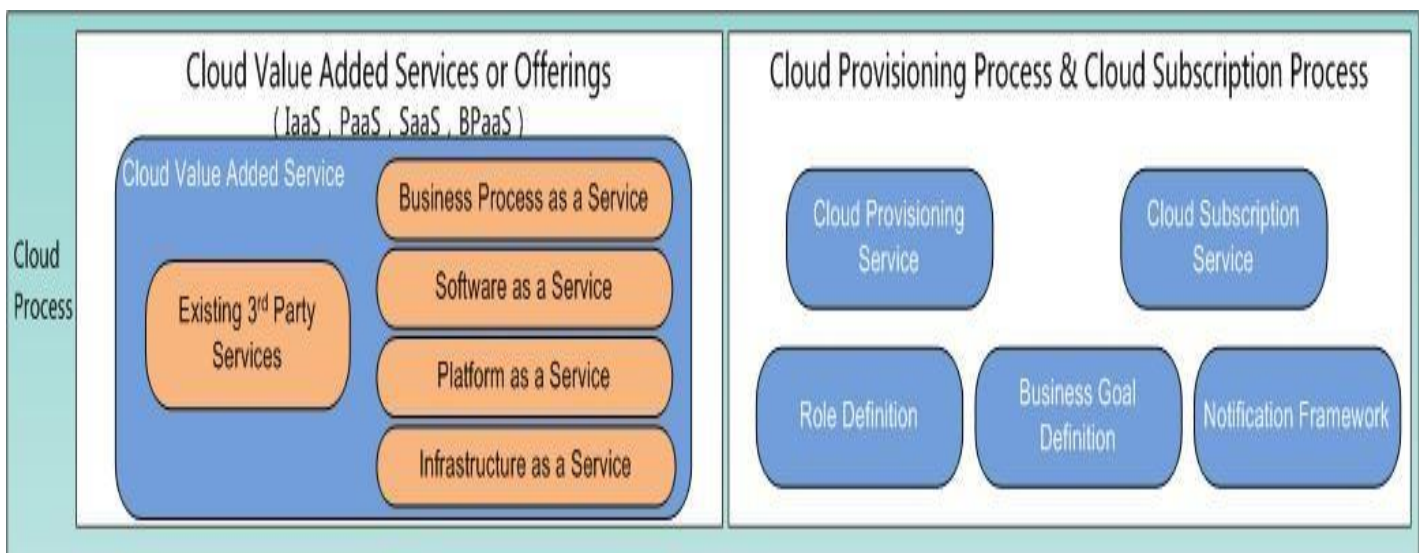


Figure 5. ABBs in Cloud Process Layer

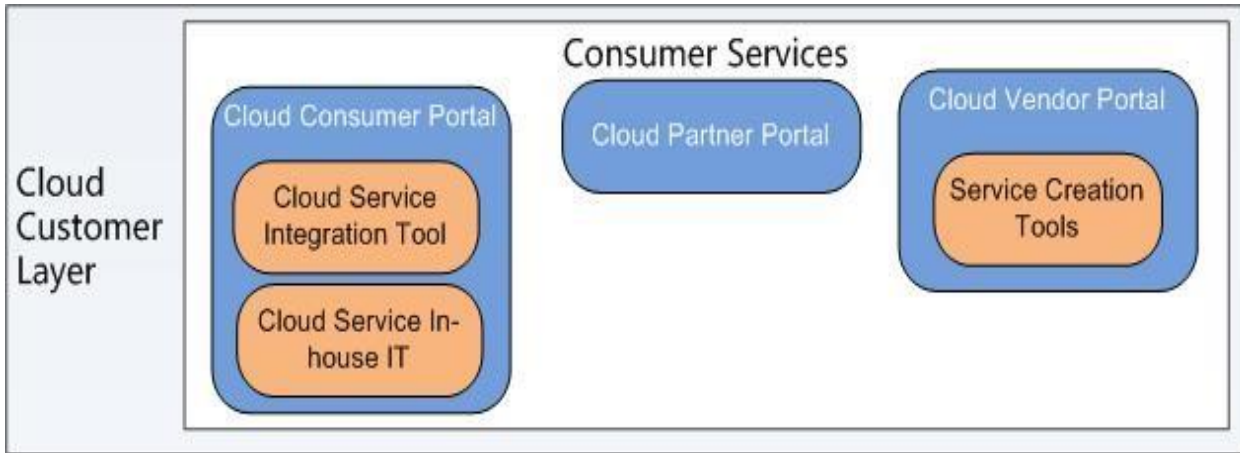


Figure 6. ABBs in Consumer Layer



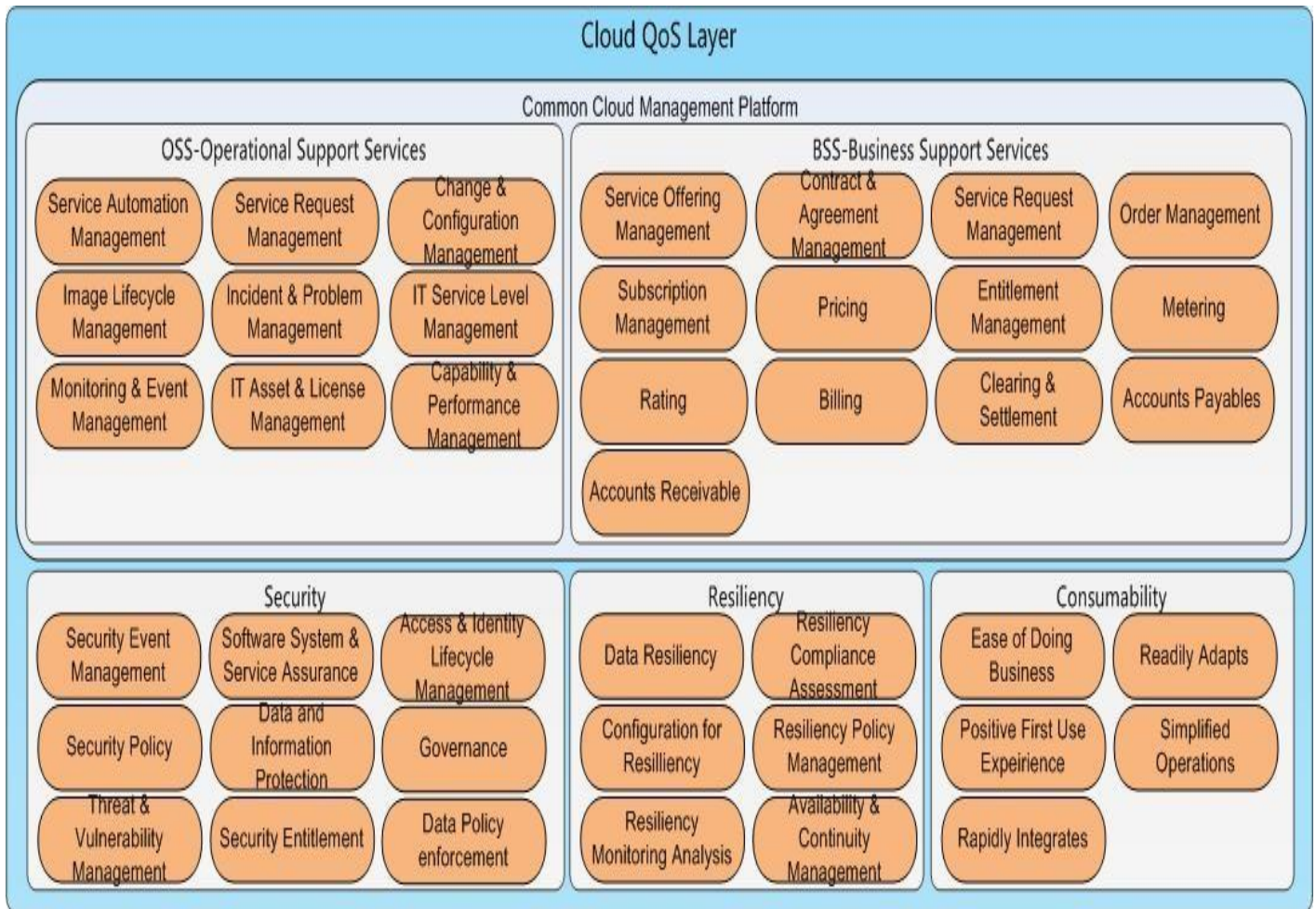


Figure 7. ABBs in Cloud QoS Layer

provisioning process to provide cloud value added services and offerings to their target cloud consumers. Cloud consumers could also use cloud subscribing process to get proper cloud value added services and offerings from cloud vendors and partners according their specific requirements. Cloud value added service or offering includes different service models of cloud computing, including IaaS, PaaS, SaaS and BPaaS.

1) *Cloud Consumer Layer:* The Customer Layer contains different portals for different Cloud Computing participants. There are three kinds of participants play important roles in Cloud Computing, including cloud vendors, cloud partners and cloud consumers. Cloud vendors expose its interaction interfaces of its internal operations and product development capability to the cloud through Cloud Provider Portal. Cloud partners provide components to cloud vendors or serve as agents to provide value-added services to the cloud clients through Cloud Partner Portal. Cloud consumers are users of the cloud services that offer business goal driven resource sharing through Cloud Consumer Portal.

2) *Cloud Integration Layer:* Message exchange of Cloud Computing resources is

very important to enable the collaborative and effective features of Cloud Computing. Cloud Computing resources include all business entities (e.g. cloud clients, partners, and vendors) and the supporting resources such as virtualization related modules, service-orientation related modules, cloud core, and cloud offerings.

In CCRA, Cloud Integration Layer is responsible for the definition of message format exchanged among cloud entities, and the description of the message exchange patterns so that they can be reused effectively. Other capabilities covered in the Cloud Integration Layer include the message routing and exchange protocols as well as message transformation capability.

- 1) *Information Layer:* Information layer is responsible for enabling representation of the cloud entities (business entities and supporting resources) in a unified Cloud Computing entity description framework so that the Cloud Computing resources can be used effectively by the collaborators. In CCRA, the Cloud Information Layer is where the unified description framework is defined.
- 2) *Cloud Quality Layer:* The Cloud Quality layer is responsible for ensuring the quality of cloud service by serving as a collection point for the administration and control, or monitoring and management of most if not all of the non-functional requirements, including security, availability, all configuration, monitoring and management capabilities and etc. In this reference architecture, all cloud related monitoring and management capabilities are realized in this layer in the Cloud Computing Management Platform(CCMP), which reflects the best practice of related projects that IBM developed.

In CCRA, the CCMP is defined as a general purpose cloud management platform which can be used by cloud participants to support the monitoring and management of cloud services across I/P/S/BPaaS. The functionalities of CCMP is encapsulated in the form of services focusing on business and operational management respectively, namely the Operational Support Services (OSS) and Business Support Services (BSS). These BSS & OSS functions must be exploited by Cloud Services to run within the context of the respective cloud service provider (and the corresponding CCMP installation).[10]

*Cloud Governance Layer:* The layer of Cloud Governance is responsible for the identification and

indicators for Cloud Computing environment and a set of normative guidance to govern the design, deployment, operation, and management of the cloud offerings.

The governance layer provides a central point in which policies are set into registries and repositories. In general, governance capabilities and processes are administered and run centrally via this layer. Note again that this layer is the underlying layer for all of the other

layers within the architecture relates to and touches upon all other functional and cross cutting layers of the foundation.

Besides these capabilities it inherits all the capabilities and ABBs from the Business Process Layer of SOA RA, including the Command and Control Management, Security Management, IT Systems Monitoring and Management, Application and SOA Monitoring and Management, Business Activity Monitoring and Management, Event Management, Policy Monitoring and Enforcement, Configuration and Change Management, Data Repository capabilities and their corresponding ABBs.

New ABBs have also been added in this layer to support the specific characteristics of Cloud Computing, such as Multi-tenancy and Interoperability.

The capabilities supported include:

- Cloud Service Registry & Repository

This category of capabilities provides ability to register and store service information.

- Interoperability Management

This category of capabilities provides ability to support services discovery across different registries and described by different semantic models.

- Service Billing Management

This category of capabilities provides the ability to bill the consumers for the services they use.

- Cloud Ecosystem Management

This category of capabilities provides the ability to manage lifecycle, evolution, interaction, relationship between the involved participants. Services and Resources are entities. The management of the user includes the management of user information of Cloud Service Consumer, Cloud Service Provider, and Cloud Service Vendor, and the interrelationship information between the three communities. User Account Management needed in Multi-tenancy is also supported. The management of the entity includes lifecycle management of services and resources, keeping record of their entire existence from creation to perish, their usage information, their interactions, evolutions and collaborations.

## I. CASE STUDIES OF CCRA

### A. *PaaS Architecture*

In this case study, we designed a concrete architecture based on CCRA. The scenario is to implement PaaS for a traditional software vendor, so that they can migrate their products to the cloud as quickly as possible. Figure 9 depicts the instantiated architecture. It is mainly composed by six parts, including the infrastructure platform, App Engine, App Platform, developer's endpoint, end user's endpoint and operating console.

The infrastructure platform is used to virtualize physical infrastructures into their virtualized counterparts, and management these virtualized resources so that they are dynamically elastic according the requirements.

App Engine is bulid on top of the infrastructure plat- form for cloud service developers. It can also be called as develop platform in the sense that it is used to provide developing environment and tools to develop enterprise level applications and host them in a simple and convenient way. App Engine consists of a series of platform services, app servers and several develop tools. Platform service provides functionalities for cloud services developing. There are two types of platform services, which are horizontal service and vertical service. Horizontal services are domain independent services used to provide some basic and common functions for developing applications, such as payment gateway for online payment, mobile gateway for short message servic- ing and so on. Different from horizontal services, vertical services are another type of platform services, these ser- vice provides functionalities for developing domain specific service, such as express tracking service based on logistics industry service, and invoice authenticity service based on retail industry service. In order to make the architecture more stable and avoid vendor lock-in, all of the platform services, including both horizontal service and vertical service are stateless and loosely coupled. Once a cloud service has been developed, cloud service developers could execute the executable program in the app server, where robust runtime environment for applications is provided.

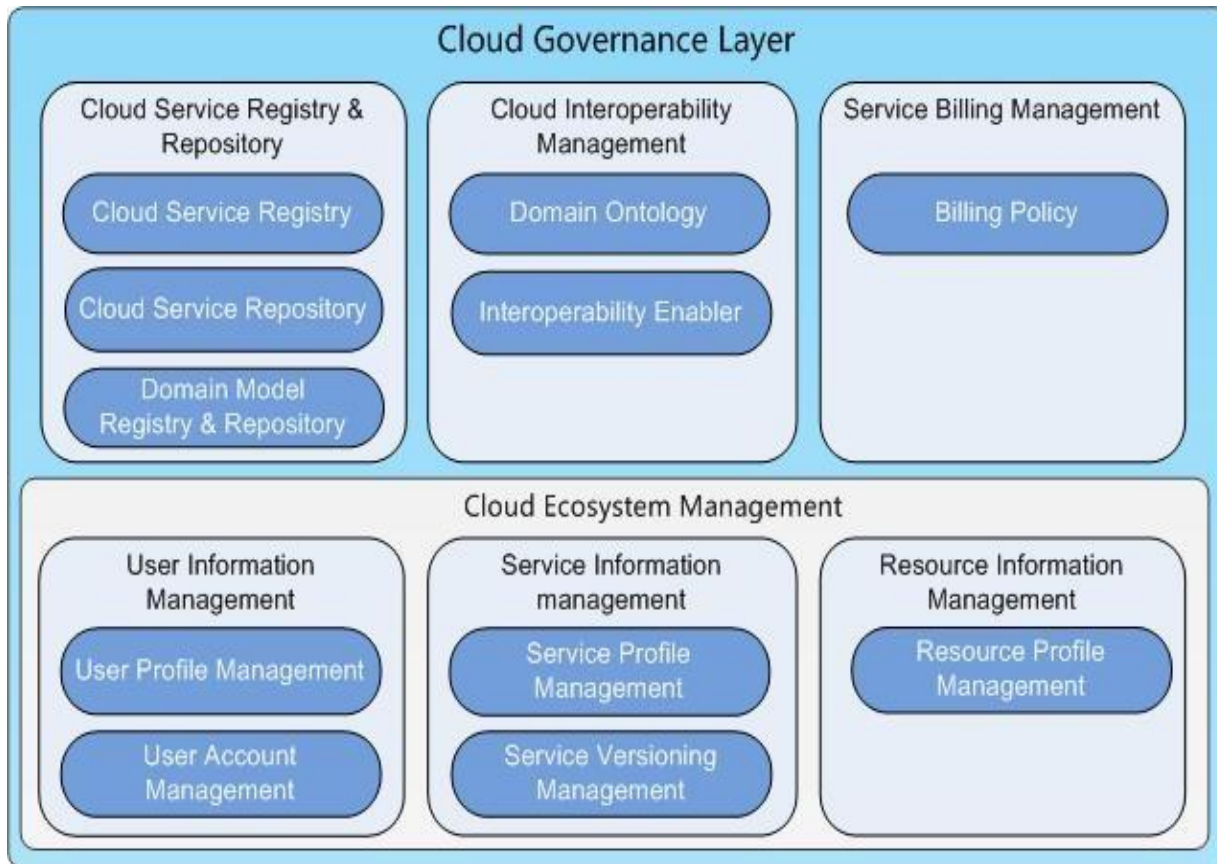


Figure 8. ABBs in Cloud Governance Layer

App platform is used for enterprises and end users in these enterprises. Enterprises order application to their cloud home from the cloud store and pay for their orders. End users could access their enterprises' cloud home to select the appropriate applications on their own app desktop for free, and they also could get other cloud service they would like to use from app store directly, but they must pay for their selections.

Developer endpoint provides IDEs, frameworks and APIs for cloud service developing. Because of the openness of the platform, developers could be every business unit of the software vender, or partners and other ISVs. These developers could use IDEs, frameworks and APIs to develop and deploy cloud services based on app engine, they could also submit them to the app store in the app platform.

End users can access end users point through different devices. These devices could be traditional pc, or mobile devices such as cell phones and tablets.

Operating console is used by cloud operators, through which they could manage enterprise and end users accessed in the cloud, and applications developing and running in the cloud.

Here we have implemented the above architecture in our projects which name SME cloud (cloud.kingdee.com), one of the central grant funded Cloud Computing demonstration projects of China. In this project, we followed the architecture to construct an open platform for ISVs, to establish a SaaS ecosystem for SMEs. It has been found that, the open

platform and its architecture are available and scalable enough, and till now, there are more than 200 applications have been developed and deployed on the platform. Take kuaidi100 ([www.kuaidi100.com](http://www.kuaidi100.com)), one of a typical application runs on the platform as an example, the average page view of the first quarter of 2012 is nearly 5 million per month.

### *B. SaaS Architecture*

In the second case study, we will use a SaaS application as an example of how proposed CCRA is applied, the instantiated architecture, as well as the typical processes that illustrate the dynamic interactions between the ABBs.

The scenario of the case is as follows. A company who has been very successful in Travel Planning Business has expanded its business to different countries in the world. Now it needs to transform its original Travel Planning System to the Cloud Computing Platform so that the system can cope with the different requirements on both business process and services that are resulted from the multinational business.

Although there are differences, the overall business processes are very much alike in different countries, mainly composed of submitting customer query, checking on the hotel and flight information, submitting payment information, making the payments. In some countries, before final step of payment, a validating procedure on the credit information is required by the local law for the payment to be effective. The validating procedure, as well as the hotel and flight checking services are provided by the third parties.

For companies which try to put their own software or Business Process on the Cloud Computing platform, the ABBs that are needed are shown in Figure 10. It covers all scenarios of SaaS and Business Process as a service (BPaaS) in the Cloud Computing environment.

The new system should support multi-tenancy. Both B-PaaS and SaaS should be customizable according to the specific needs of its tenants. CCRA is used to separate the concerns of business execution and supporting infrastructure. When the customer submits his information, the system will call the respective business process or the third-party services according to the tenant he belongs to. For those tenants who need credit validation in their business process, credit validation services will be called. Different hotel and flight information retrieval services from different service providers are called accordingly as well. These services are implemented and executed by third parties. From the business model perspective, third party services to be used internally or as a value-added cloud offering for other tenants to subscribe.

The Traveling Planning System leverages the APIs of these third party services together with the services formed by the services encapsulated from component of its own making in the Cloud Consumer Portal.

Once the company successfully realize its Travel Planning SaaS system, there are mainly two typical processes involved in the use of the SaaS application. One is the Service Subscription Process which is triggered whenever new customer use the system to locate the



the service according to his own specific demand. The other is the Service Provisioning Process, which is triggered when new services are provisioned. The two processes are described as follows.

1) *SaaS Service Subscription Process*: As shown in Figure.11, Cloud Consumer access the Cloud Consumer Portal which employs Cloud Subscription Service block ( 1 ) uses Process Flow Manager( 5 ) to select and orchestrate services from the Cloud Service Registry ( 6 ) based on Role ( 2 ) and Business Goal definition ( 3 ). If no service is located in the local Cloud Service Registry, Interoperability Enabler is used to locate services in other registries with different semantic models ( 4 ).

In the process of service discovery, role and business goal are very useful to bridge the gap between business and IT. By including the information in the Cloud Service Registry, the precision and efficiency of service query will be improved. When new service is created or provisioned in the Cloud

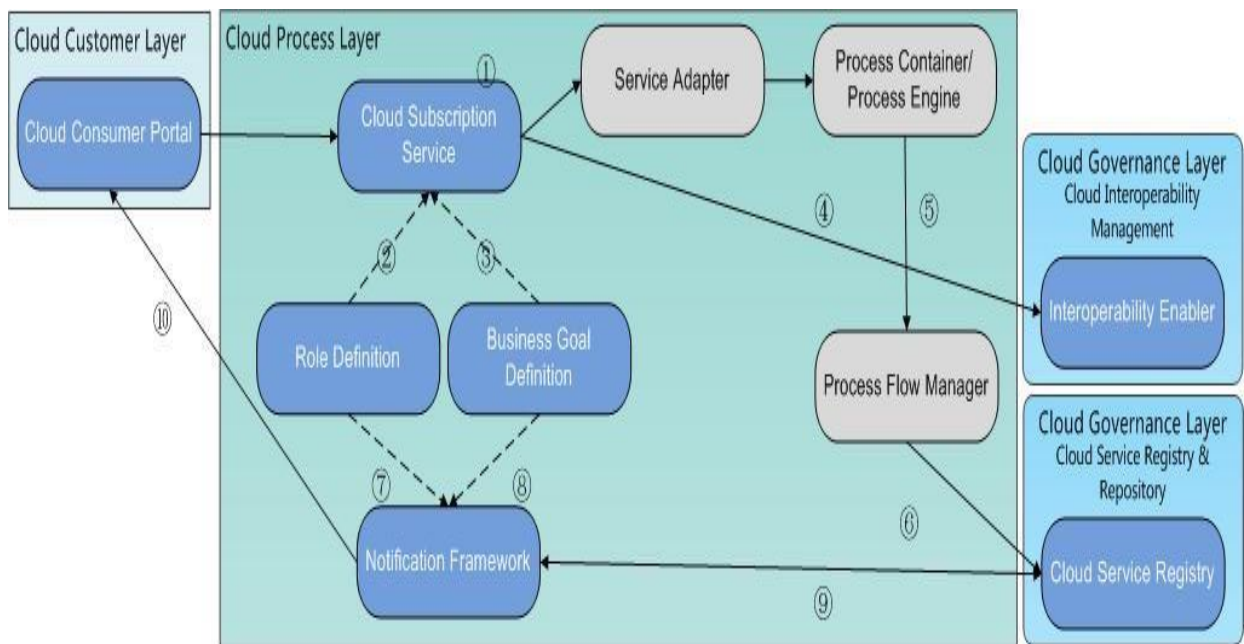


Figure 11. Interaction of Subscription Process and other ABBs

Service Registry, Notification Framework will deliver the up-to-date service information to Cloud Consumers according to their interests defined by roles( 8 ) and business goals( 7 ).

1) *SaaS Service Provisioning Process*: Through Cloud Vendor Portal, Cloud Vendor is able to provision services with the help of Cloud Provisioning Service ( 1 )(Figure.12). ABBs, such as Service Adapter, Process Controller, Process Flow Manager, inherited from SOA RA are used to provision service, according to the demand of Cloud Consumer( 2 - 7 ). Each provisioned service is associated with particular Role and Business Goal definition.

Therefore, the common reusable services from the Cloud Service Layer can be selected and orchestrated by Process Flow Manager according to their specific role and business goal.

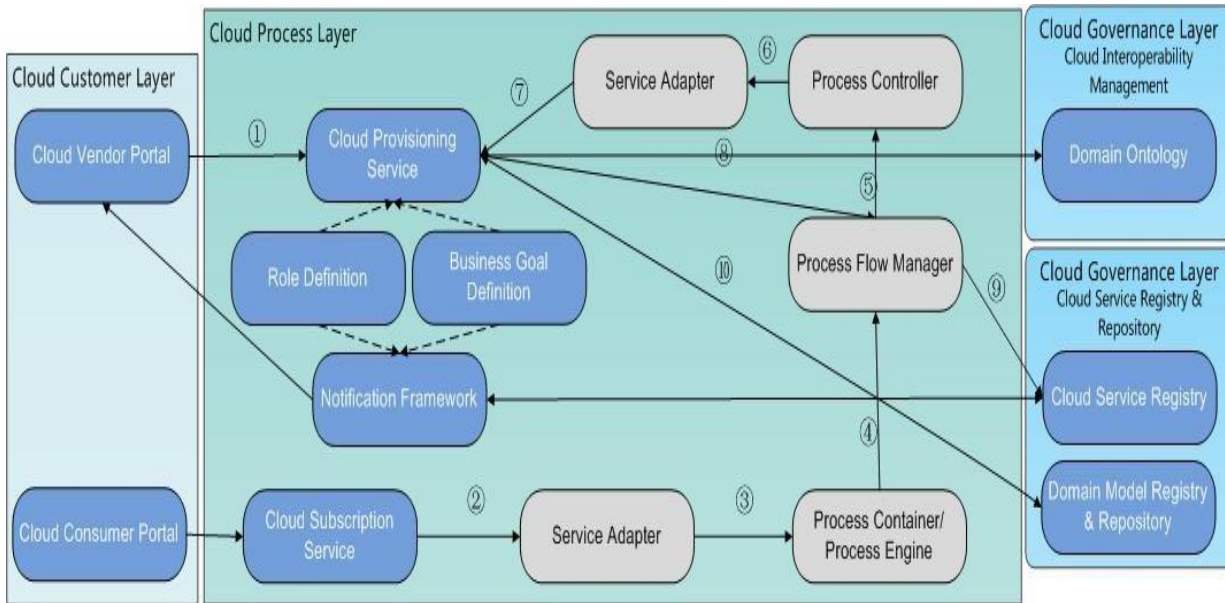


Figure 12. Interaction of Provisioning Process and other ABBs

If the provisioned service is new, Cloud Service Registry is updated to include the new service information, new domain models and semantic models of the service are registered respectively into the Domain Model Registry & Repository (10) and Domain Ontology ( 8 ). The service information can thus be used by other Cloud Provisioning Services or Cloud Subscription Services.

Notification Framework will deliver the up-to-date service information to Cloud Partners according to their interests defined by roles and business goals.