Building the Cloud Infrastructure

Cloud computing reference model

What is a reference Model

A reference model is an abstract framework for understanding significant relationships among the entities of some environment, and for the development of consistent standards or specifications supporting that environment. It is based on a small number of unifying concepts and may be used as a basis for education and explaining standards. It is not directly tied to any standards, technologies, or other concrete implementation details, but it does seek to provide a common semantics that can be used unambiguously across and between different implementations. - **Organization for the Advancement of Structured Information Standard (OASIS)**

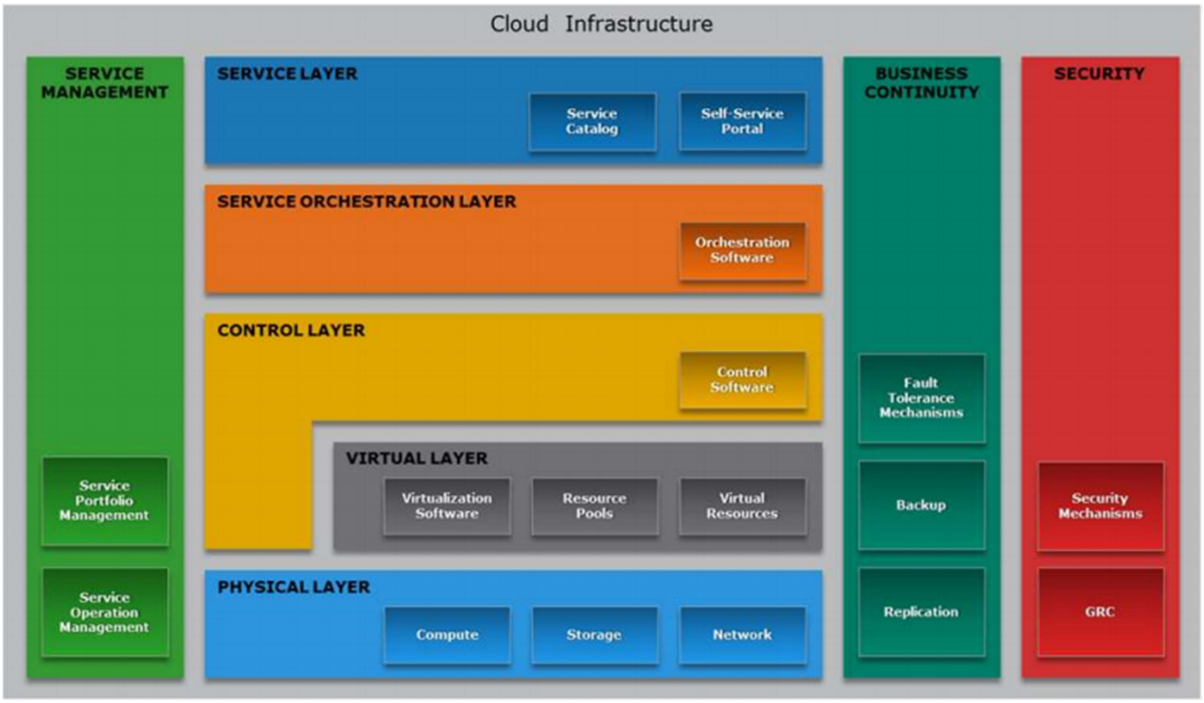
It facilitates efficient communication of system details between stakeholders

It provides a point of reference for system designers to extract system specifications

Key goals of reference model are:

- Conveys fundamental principles and basic functionality of a system it represents
- Facilitates efficient communication of system details between stakeholders
- Provides a point of reference for system designers to extract system specifications
- Enhances an individual's understanding of the representative system
- Documents the system for future reference and provides a means for collaboration

Cloud Computing Reference Model



The cloud computing reference model is an abstract model that characterizes and standardizes the functions of a cloud computing environment by partitioning it into abstraction layers and cross-layer functions.

This reference model groups the cloud computing functions and activities into five logical layers and three cross-layer functions.

The five layers are physical layer, virtual layer, control layer, service orchestration layer, and service layer.

Each of these layers specifies various types of entities that may exist in a cloud computing environment, such as compute systems, network devices, storage devices, virtualization software, security mechanisms, control software, orchestration software, management software, and so on.

It also describes the relationships among these entities. The three cross-layer functions are business continuity, security, and service management. Business continuity and security functions specify various activities, tasks, and processes that are required to offer reliable and secure cloud services to the consumers.

Service management function specifies various activities, tasks, and processes that enable the administrations of the cloud infrastructure and services to meet the provider's business requirements and consumer's expectations.

Deployment options and solutions for building cloud infrastructure

Before building a cloud infrastructure, organizations must identify which deployment option is appropriate for them. There are two deployment options for building a cloud infrastructure and they are Greenfield deployment option and brownfield deployment option.

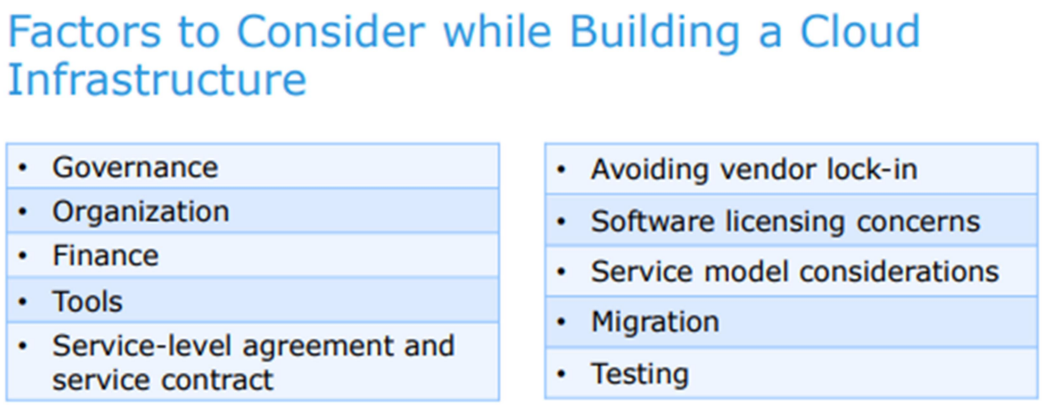
A Greenfield deployment option is typically used when an infrastructure does not exist and an organization ha to build the cloud infrastructure starting from the physical layer. On the other hand, a brownfield deployment option is used when some of the infrastructure entities exist, which can be transformed to a cloud infrastructure by deploying the remaining entities required for the cloud infrastructure. For example, consider that an organization wants to use a brownfield deployment option to transform their existing data centre, which has the physical, virtual, and control layers deployed. In such cases, the data centre also has the business continuity, security, and service management in place. However, these three cross-layer functions are limited to a non-cloud environment. While transforming the existing data center to a cloud infrastructure, the organization will have to deploy the orchestration layer and the service layer. Further, the BC, security, and the service management functions will have to be transformed to support the cloud environment. In both deployment options, apart from deploying the five layers and the three cross-layer functions, the organizations have to consider several factors that will enable them to deploy the cloud services that will meet the consumers' expectations

There are two solutions for building a cloud infrastructure: by integrating best-ofbreed cloud infrastructure components and by acquiring and implementing a cloudready converged infrastructure.

In an integrated best-of-breed cloud infrastructure components solution, organizations have the flexibility to use and integrate the infrastructure components from different vendors. This solution allows organizations to design their cloud infrastructure by repurposing their existing infrastructure components (in a brownfield deployment option), providing a cost advantage for this solution. This solution enables organizations to select a vendor of their choice for infrastructure components. This solution also enables an organization to easily switch a vendor if the vendor is unable to provide the committed support and not meet the SLAs. When this method is used to build a cloud infrastructure, an organization may have to spend a significant amount of IT staff time evaluating individual, disparate hardware components, installing hardware, and integrating compute, storage, and network components. The IT staff may also have to spend effort integrating and testing hardware, middleware, and software. They also need to check the compatibility of all the components to ensure that the combined components interoperate and function as expected. This may delay the deployment of cloud services. Further, scaling of such an infrastructure takes longer because each component that is scaled requires integration with the existing infrastructure and testing for compatibility. Finally, this solution requires acquiring cloud infrastructure management tools and deploying them on the infrastructure.

A cloud ready converged infrastructure solution provides a modular design that combines compute, storage, network, virtualization, and management components into a single package. This package is a self-contained unit that can be utilized to deploy cloud services, or can be aggregated with additional packages to support the demand for more capacity and performance. The package is pre-configured, reducing the time to deploy cloud services. Further, in addition to integrating various components into a package, this solution offers single management software capable of managing all hardware and software within the package. A cloud-ready converged infrastructure solution has built-in capabilities that provide secured multi-tenancy. However, additional security mechanisms should be deployed to prevent external attacks. The solution is capable of managing and mitigating failure scenarios in hardware, software, and cloud services. A potential area of concern regarding cloud-ready converged infrastructure solutions is the lack of flexibility to use infrastructure components from different vendors. Some vendors may provide organizations with the flexibility to choose multi-vendor infrastructure components such as network devices, compute systems, and hypervisors for this solution.

Considerations for building cloud infrastructure



After deciding on the deployment option and solution to build the cloud infrastructure, a cloud service provider have to consider several factors to deliver cloud services that meet their business objectives and consumer's expectations.

Governance

Governance is the active distribution of decision-making rights and accountability among different stakeholders in an organization. It also describes the rules and procedures for making and monitoring those decisions to determine and achieve the desired behaviors and results. The role of governance in IT is to implement, maintain, and continuously improve the controls on the use of IT resources. IT governance enables a service provider to:

• Ensure that IT resources are implemented and used according to agreed-upon policies and procedures

• Ensure that these resources are properly controlled and maintained

• Ensure that these resources are providing value to the organization

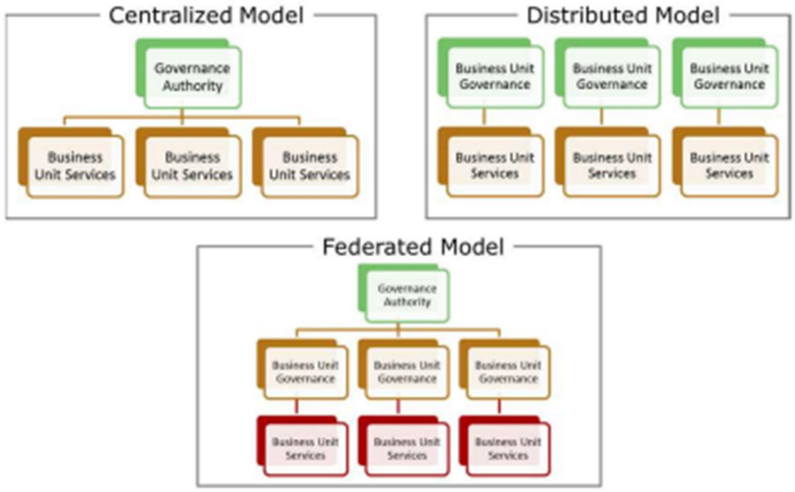
Instituting IT governance usually involves establishing a review board, which is a team of members from across business units including IT. This review board is responsible for creating rules and processes that the organization must follow to ensure that policies are being met. These rules and processes might include the following:

• Understanding business issues, such as regulatory requirements or funding

• Establishing best practices and monitoring these processes

• Assigning responsibility for things such as standards, design, review, and certifications If a service provider is using a greenfield deployment option for building a cloud infrastructure, then they must establish governance by choosing appropriate governance model. If a service provider is using a brownfield deployment option, then that service provider must transform their existing governance model to meet the cloud requirement.

Governance Models



Depending on the size, structure, geographic presence, and culture of an organization, one of these fundamental governance models can be implemented:

• A centralized model provides one governance body for the entire organization. This fits best with a smaller or a strongly centralized organization where governance policies are, for the most part, consistent throughout the organization.

• A federated model proposes separate governance bodies, one for each business unit. A business unit can be a functional organization, a product group, or a geographic location. Each business unit has its own set of governance policies. Even though the services for a given business unit can be independently standardized, managed and owned, a single, enterprise-wide governance body can still subject all services to a common governance system.

• A distributed model proposes separate governance bodies for each business unit. These governance bodies function autonomously and are not controlled by any common governance system. The organization can choose a governance model that best meets its requirements. After a governance model is chosen, the organization then needs to take steps to establish or transform to the chosen governance model.

Organization

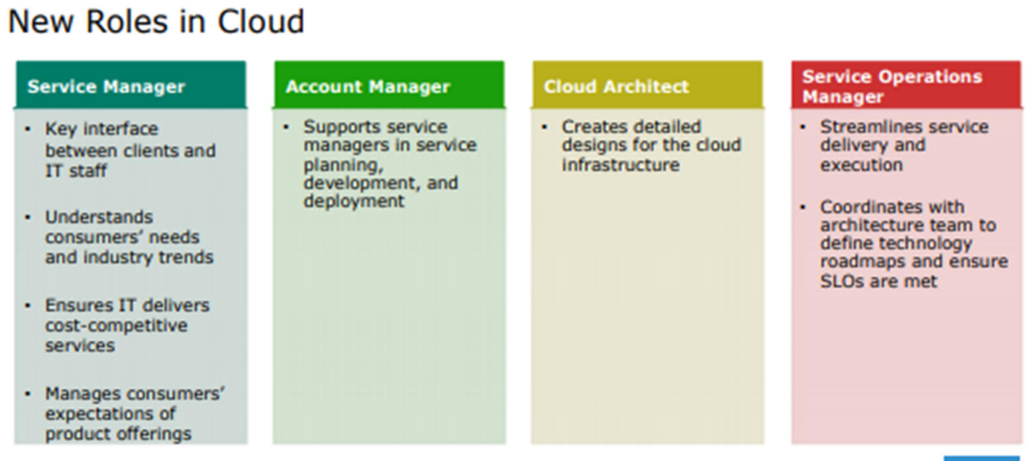
A cloud service provider needs to institute or transform the organization to a proactive and services-based model. This requires defining several new roles that perform tasks related to cloud services, such as service definition and creation, service administration and management, service governance and policy formulation, and service consumer management. Some of these tasks can be combined to become the responsibility of an individual or organizational role. A few examples of new roles required to perform tasks within a cloud environment include service manager, account manager, cloud architect, and service operation manager.

• A service manager is responsible for understanding consumers' needs and industry trends to drive an effective product strategy. The service manager ensures that IT delivers costcompetitive services that have the features that clients need. The service manager is also responsible for managing consumers' expectations of product offerings and serves as key interface between clients and IT staff.

• An account manager supports service managers in service planning, development, and deployment. The account manager maintains day-to-day contact to ensure that consumers' needs are met.

• A cloud architect is responsible for creating detailed designs for the cloud infrastructure.

• The service operations manager is responsible to streamline service delivery and execution. Service operations manager is also responsible to provide early warning for service issues, such as emerging capacity constraints, or unexpected increase in cost. The service operations manager also coordinates with the architecture team to define technology roadmaps and ensure that service level objectives are met.



It is not only required to define the roles of IT staff and the skills they need, but it is also essential to identify the skill gaps that should be filled in order to successfully provide cloud services.

Finance

A service provider needs to institute or transform the financial/payback/show back/pricing model that will enable them to manage their budgeting, accounting, and chargeback requirements. The model helps the service provider to plan for investments to offer cloud services and determines the IT budget for cloud infrastructure and operations for the lifecycle of services. The service provider should perform service valuation. Service valuation determines the price (or chargeback) that a consumer is expected to pay for a service, which helps recover the cost of providing the service, ensuring profitability, and meeting the provider's ROI and reinvestment goals. The service provider aggregates all types of costs (both CAPEX and OPEX) down to the service element level of granularity by mapping the elements to the relevant cloud services. Then it calculates the service costs on per-unit basis by dividing the aggregated cost for a service by some logical unit of demand such as GB of storage or an hour of usage for that service.

However, the per-unit service costs may vary over time, depending on the demand for or utilization of the services and service elements. Thus, service provider should track the demand and utilization to establish a stable per-unit cost baseline. Finally, the service provider may add some margin amount over per-unit service cost to define service price, or may establish the price at the true cost of service depending on the provider's business goal.

The service provider then defines chargeback or showback model(s) based on the pricing strategy for cloud services.

A chargeback model defines how consumers need to pay for the consumed services. A list of common chargeback models along with their descriptions are provided below. • Pay-as-you-go: Metering and pricing is based on the consumption of cloud resources by the consumers. Consumers do not pay for unused resources.

• Subscription by time: Consumers are billed for a subscription period. The cost of providing a cloud service for the subscription period is divided among a predefined number of consumers. For example, in a private cloud, if three business units are subscribing to a service that costs $60,000 a month to provide, then the chargeback per business unit is $20,000 for the month.

• Subscription by peak usage: Consumers are billed according to their peak usage of IT resources for a subscription period. For example, a provider may charge a consumer for their share of peak usage of network bandwidth.

• Fixed cost or pre-pay: Consumers commit up-front on the required cloud resources for the committed period such as one year or three years. They pay fixed charge periodically through a billing cycle for the service they use, regardless of the utilization of resources.

• User-based: Pricing is based on the identity of a user (a person) of cloud service. In this model, the number of users logged in is tracked and billed, based on that number.

Service provider deploys chargeback tools in the cloud infrastructure. These tools enable service provider to define a chargeback model. Based on the model, these tools automatically collect billing data, store billing records in a billing system, and generate the billing report per consumer.

Tools

Tools play an important role in building cloud infrastructure; therefore an early step in building the infrastructure is to deploy the necessary technologies using the tools. Examples of the key tools used for building cloud infrastructure include virtualization software, orchestration software, security software, business continuity software, self-service portal software, and so on. These tools enable the service provider to build and offer cloud services to the consumers.

Apart from considering the tools that enable the providers to build a cloud infrastructure, providers should also consider tools that will enable them to connect multiple clouds or applications. Examples of such tools include cloud integration tools, APIs, and specialized connection, transformation, and business logic programs. These types of tools are specially useful while deploying hybrid or community cloud.

Also, such tools are important to consider when a service provider is providing brokerage services. Cloud integration tools enable connecting cloud applications with other cloud and non-cloud applications to leverage the capabilities of multiple applications. Cloud integration technology integrates multiple cloud applications using application programming interface (API) support. These APIs enable secure access to the data of integrated applications. However, integration cannot be accomplished only with APIs because they do not perform functions such as transformation of data formats, data mapping, data validation, and error processing. These functions are typically handled by specialized connection, transformation, and business logic programs. These programs gather data with the help of APIs, then transform formats as required, and validate the accuracy of the transformation.

Consumers may avail different cloud services from multiple providers. In such cases, consumers may need assistance in selecting the providers that best meet their requirements. Moreover, using multiple cloud services from different providers may lead to operational complications and integration issues between the various services.

Such issues have led to the emergence of cloud consumption assistance services known as cloud services brokerage, which are provided by cloud brokers.

Service-level Agreement and Legal Contract

A service-level agreement (SLA) is a contract negotiated between a provider and a consumer that specifies various parameters and metrics such as cost, service availability, maintenance schedules, performance levels, service desk response time, and consumer's and provider's responsibilities. SLAs must be carefully written before offering to a consumer.

SLAs are part of a service contract: an agreement between the cloud service provider and the cloud service consumer, stating the terms of service usage. A legal contract must be established with the consumer before a service can be used.

When writing a legal contract, the key considerations include business level policies such as data privacy, data ownership, data retention, secure deletion, security, confidentiality, auditing, regulatory requirements, redundancy, jurisdiction, disruption resolution, compensation for data loss and misuse, excess usage, availability and performance metrics, payments and penalty methods, contracted services, a list of services not covered, licensed software, and service termination.

Finally, a disaster recovery plan, penalties, and an exit clause should be included. An SLA should include an indication of how an unexpected incident will be handled and what actions will be taken in case of a prolonged service outage.

It should cover penalties for not meeting the SLA. The SLA should also include clauses related to the termination of the service by both the consumer and the provider.

Avoid Vendor Lock-in

Cloud vendor lock-in refers to a situation where a consumer is unable to move readily from the current provider to another. This condition may result from various causes such as high cost of migration, significant re-engineering effort requirement for an application migration, lack of open standards, or restrictions imposed by the current provider.

When building a cloud infrastructure, providers must avoid using proprietary tools, APIs, or file formats, which may cause vendor lock-in. The use of widely accepted open standard tools, APIs, and file formats not only prevent vendor lock-in, but also make services offered using open tools more acceptable to the consumers. The use of open standards provides interoperability and portability among providers, which consumers typically prefer. For example, the provider may use APIs based on the open standards that enable an application's data to migrate to another provider with minimal or no change to its format.

Likewise, if the provider supports the use of Open Virtual Machine Format (OVF), which is an open standard for virtual machine format, then a virtual machine created in one of the provider's environment can be migrated to another provider with minimal or no changes.

Sometimes providers may impose restrictions or burdensome penalties for migrating to another provider, causing lock-in. Including an appropriate exit clause in the SLA can prevent vendor lock-in due to restrictions and penalties

Software Licensing Concerns

While building a cloud infrastructure, providers must consider challenges associated with software (application and operating system) licenses. It is important to asses these challenges at an early stage. Software licensing challenges are relevant to infrastructure as a service (IaaS) and platform as a service (PaaS) models.

Consumers can use their existing software license in the cloud only if it is cloud enabled. Therefore, providers must identify whether the consumer's existing software license is cloud enabled. If not, then the consumers can pay additional fees to get their license cloud enabled.

Alternatively, consumers can use the software provided by the service provider and pay a fee for the software usage.

Further, the service provider in collaboration with the software vendors and consumers must work to understand the software license rights and its usage. This is important because the cloud service provider may have to create redundant systems by replication to combat against unplanned outage or disasters. Understanding the license rights and its usage will enable the service providers preventing any non-compliance and violation of the license agreement.

Service Model Considerations

Considerations for SaaS

• Software as a Service: – Ensures the software offered are thoroughly tested

– Ensures the new features and functionalities are developed to the software to meet consumer's needs

– Ensures applications are scalable and can handle increasingly larger consumer workloads

– Ensures the applications are resilient and can withstand failures such as

- Underlying component failure
- Dependent service failure

– Ensures the consumers are provided a secure environment

Considerations for PaaS and IaaS

• Platform as a Service:

– Provides application development platform to the consumers

– Supports large variety of OS, application development tools, and deployment tools

– Ensures the consumers are provided a secure environment

– Provides the consumer the required computing resources to operate the application

• Infrastructure as a Service:

– Provides the consumer the required infrastructure resources to deploy their OS, application, and data

– Ensures that the consumers are provided a secure environment

Migration

Migration strategy and considerations depend on whether a consumer plans to migrate their application (in case of IaaS) or only their data (in case of SaaS). For application migration, service providers must work with consumers to develop a migration strategy for their application.

Also, they must identify the various dependencies of the application. For example, if an application depends on an authentication service that is on-premise, then appropriate configuration changes are required in order to make the application work after migrating to a cloud. Based on dependencies, a consumer may choose one of the two migration strategies.

The strategies are forklift migration and hybrid migration.

• In the forklift migration strategy, the application and all of its related components are migrated to the cloud at once. This strategy is typically used for tightly coupled applications or self-contained applications.

Tightly coupled applications are multiple applications that are dependent on each other and cannot be separated. Selfcontained applications are applications that can be treated as a single entity.

• In a hybrid migration strategy, an application and its components are moved to the cloud in parts. This strategy is a lower-risk approach to migrate applications to the cloud. This is because parts of an application can be moved to the cloud and optimized before moving other parts.

This reduces the risk of unexpected behavior of the application when it is moved to the cloud. This strategy is typically good for applications with many loosely coupled components. In some cases, consumers may only require migration of data.

The data can be migrated to the cloud by deploying replication technology to copy the data from the consumer's data center to the cloud.

While migrating data to the cloud, the provider must consider the factors such as network bandwidth, data security, data integrity, data consistency, jurisdiction, and so on.

Testing

After the application or data is migrated to the cloud, the provider must work with the consumer to test their application to ensure that it is working as expected. The degree of testing may vary depending on the scope and magnitude of the consumer's requirements. While developing a test strategy, the providers in collaboration with the consumers must consider the following:

• Define roles and responsibilities of the personnel involved in test and quality assurance (QA) process

• Identify the tools required to perform test management and automation

• Design tests for data migration to the cloud

• Design test cases to perform various testing modes such as stress, performance, functional, interoperability, and compatibility

Apart from testing the application, the provider must also test other cloud capabilities such as fault tolerance, disaster recovery, security controls, and any other capabilities to ensure that the migrated application has successfully been configured with the capabilities that are committed by the provider.