

Cloud solutions

Cloud application planning

Design and development of cloud apps requires many considerations:

- Business functions
- Application architecture
- Security
- Cloud delivery models
- User experience
- Development , testing and runtime environment

Business Support Services(BSS)

- Applies to service providers
- Are components that cloud operators use to run the business operations
- Activities involved are taking customer orders, managing customer data, managing order data, billing, rating, and offering services

Operational support services(OSS)

- Are computer systems used by service providers
- Includes network systems, support processes such as maintaining n/w inventory, provisioning services, configuring network components and managing faults

Cloud business process management

- It governs an enterprise's cross functional, customer focussed, end to end core business process.
- It achieves strategic business objectives
- It drives the resources across the organization to create customer value
- It focuses on integrating verticals and optimizing core work

Cloud environment can help in

- Integration of core business
 - Cross organizational functions
- Value focused
 - Customer centric perspective
 - Performance measurement
- Continual improvement
 - This is based on longer time of intervals pertaining to cloud business

Cloud technical strategy

- Enabling support for multi tenancy
- Self service registration
- Managing customers and their entitlements
- Single sign-on
- Security concerns
- Manage firewall, recovery and backup issues

Cloud use cases

IAAS or test/development

Problem: Dev team requires unpredictable amounts of infra to do job. Procuring and setting up resources can be challenging.

Solution: IaaS

Standardized development platforms(PaaS)

Problem: developers are not concerned about the impact of code on architectural decisions.

As many developers involved there can be non standard development platforms.

companies sometimes create their own standards around development stacks that include apps.

Solution: PaaS

companies can create standardized dev platform definitions for use across dev and test teams.

This improves productivity.

Cloud service management system

- Provides visibility, control, and automation needed for delivery.
- It requires
 - Simple UI
 - Lowering cost with provisioning
 - Increase system administration productivity
- Cloud brokers
 - Third party to manage the specialized cloud service providers
 - Such as building services on top of existing cloud platform
 - Deploying services over multiple cloud platforms.

Key characteristics of orchestrator and engine

- Scalability
 - Cloud Orchestrator (**CO**) maintains an index of resources acquired from the hypervisor.
 - This enables it to scale and lowers the overhead on the master server.
- High Availability
 - CO must support ***active-passive*** as well as ***active-active*** scenarios for availability and disaster recovery.
 - Must detect failures of nodes and automatically start nodes to compensate for the failure of node.

- Application lifecycle
 - CO must support application lifecycle.
 - Creating infrastructure, installing, configuring, and launching apps until deletion or expiration.
- Multi-tenancy/ role based administration
 - CO must support capability to access shared infrastructure but with different access permissions.
 - Role based administration allows fine grained control over the CO features
- Policies
 - CO must support variety of policies to be enabled
 - Policies can be global level or at restricted levels
 - Policies can take effect automatically based on metric threshold (for storage, or VMs)

- Alarms
 - To alert users of any events or information status of the app
- CO must be able to optimise allocation of resources
- Reporting and accounting
- Self service portal
 - CO must provide self service portals for app owners
 - To request machines and to monitor and control resources

On premise CO and provisioning engine

- Can be bundled offering, that includes h/w and s/w for a cloud computing setup
- It includes all element of cloud ecosystem
- Has a self service system
- Includes automation
- Should be able to track and control all resources
- It can be a *packaged private cloud offering* that combines h/w, s/w and services needed to establish a private cloud
- Accelerates selling process and effectiveness
- Provides an alternative to traditional IT infrastructure
- Should Enhance delivery of services
- Transforms data centre into a cost effective dynamic infrastructure

Benefits

- Innovation
 - Improve business value
 - Enables workloads to be efficiently deployed when and where they are needed
- Decrease operational expenses
 - Gain productive increases
 - IT labour costs reduced through automation
- Reduce complexity and Risk
 - With automation and standardization the human error factor minimised

CO and provisioning requirement analysis

- Initiate a process of discussion with the customer for cloud opportunity
- Set the boundaries of the environment
- 30-40 percent of IT environment allotted for test/development
- A CO engine should allow a developer to login to self service portal,
 - Select resources required
 - Select timeframe
 - Select an image to provision from catalogue

- CO and provisioning engine solutions are great entry point for users to get on with cloud computing
- Customer data centers (traditional) supports hundreds of distinct apps where 70 percent of expense is on assembling and reassembling existing infrastructure.
- This can be altered by using cloud and assign all the existing resources to cloud.
- This will enable efficient use of resources
- Hence the opportunities are two faced:
 - Turn existing environment into to cloud
 - Install cloud management platform and assign existing resources
 - Jump start a new cloud
 - Requires h/w + s/w + services

Cloud infrastructure security

- Security can be included as a layered approach
- Top service layer can be a runtime secure virtualised environment
 - That includes administrative and security across different clouds
 - Manage web service attacks
- secure virtualised runtime layer at the bottom can provide access to data from data stores
- As cloud infrastructure is similar to SOA, different services from different providers can be integrated that includes large number of tenants, services and standards.

Computing on demand

- Can be implement using virtualization techniques
- Leverage computing resources without having to buy h/w
- Ability to transfer workloads if resources not able to support it
- Lets others utilize resources when idle
- Using resources efficiently

CoD allows to:

- Align cost with utilization
- Increase availability
 - By moving workloads dynamically from servers when necessary with apps being available
- Balance workload dynamically across multiple servers without taking apps offline
- React to short term resource requirements instantly
- Increased system utilization
- Develop simple pricing model to reflect the usage of service

Pre-provisioning

- CoD may require pre provisioning to reduce time to commission servers
- It is ideal when:
 - Capacity planning is done
 - Workload is constant to ensure good utilization
 - Business requires physical separation of workloads
 - Workloads can be scaled horizontally

Cloud platform characteristics based on CoD

Based on low end servers:

- High admin cost due to management of more physical resources
- Limited and complex scalability
- Longer time for deployment from ordering of servers to setting up of infrastructure
- Fixed cost expenditure for h/w and hence not ideal for short lifecycle
- Wastage of h/w for apps that depend on volatile markets
- Wastage of resources if not fully utilized

Based on on-demand platform:

- Lower administration costs due to less physical servers
- Can cater to quick requirements for new app deployment
- Not ideal for short lifecycle product as h/w cost is fixed

Based on Dynamic capacity platform

- Automatically provide additional processor capacity on a temporary basis in shared processor pool.
- Dynamic allocation of workloads and not dependent on h/w or apps to dynamically move
- Choose virtual machine implementation
- Lower admin cost due to less physical and logical servers
- Can cater to quick requirement for new app deployments
- Able to share i/o , cpu and memory resources between apps
- Can make use of unused processor cycles of other apps
- Workloads can be easily transferred without forcing downtime to facilitate maintenance