

The Practice of “Architecting” Cloud Solutions

Mahesh H. Dodani, IBM, U.S.A.

1 CLOUD SOLUTIONS BASED ON ARCHITECTURE

“It’s time to accept reality. SOA fatigue has turned into SOA disillusionment. Business people no longer believe that SOA will deliver spectacular benefits. “SOA” has become a bad word. It must be removed from our vocabulary.

The demise of SOA is tragic for the IT industry. Organizations desperately need to make architectural improvements to their application portfolios. Service-orientation is a prerequisite for rapid integration of data and business processes; it enables situational development models, such as mashups; and it’s the foundational architecture for SaaS and cloud computing. (Imagine shifting aspects of your application portfolio to the cloud without enabling integration between on-premise and off-premise applications.) Although the word “SOA” is dead, the requirement for service-oriented architecture is stronger than ever.

But perhaps that’s the challenge: The acronym got in the way. People forgot what SOA stands for. They were too wrapped up in silly technology debates (e.g., “what’s the best ESB?” or “WS-* vs. REST”), and they missed the important stuff: architecture and services.” – [SOA is Dead; Long Live Services ... Anne Thomas Manes Blog](#)

We start the new year by continuing the practice of designing cloud solutions based on an architecture. This article shows how to architect a cloud solution by looking at the typical requirements inherent in any cloud implementation and showing how these requirements can be supported by the capabilities provided by the cloud architecture. We start by summarizing the cloud architecture, followed by laying down the approach that can be used to adopt cloud, and then show how the implementation of the solution can be “architected.”

As a foundation, the cloud architecture shown in Figure1 needs to support the basic requirements of any cloud solution, including:

- Delivering cloud services: where applications, data, and IT resources are rapidly provisioned and provided as standardized offerings to users over the web in a flexible pricing model.

- Managing cloud services: where large numbers of highly virtualized resources are managed such that, from a management perspective, they resemble a single large resource. These resources can be managed in such a way to facilitate rapid elasticity dependent on the demand for the cloud services.

The three main roles in this architectural model are the service consumer (left hand side), the service provider (middle) and the service creator (right hand side). The service provider hosts services which are created by the service creator, based on a management platform consisting of an Operational Support System (OSS) and a Business Support System (BSS).

For example, the role of a service consumer could be fulfilled by a software development organization consuming test environments from the enterprise data center being operated as a private cloud. As another example, the service consumer role could be an internet-connected user consuming web conferences-as-a-Service based on a web conferencing and collaboration offering.

The service creator role could be fulfilled by an organization delivering applications to be hosted in the Cloud. The role of a service creator is particularly relevant for today's Independent Software Vendors (ISVs) since cloud computing as a new delivery model for IT services will dramatically change their existing business and license models (software is not necessarily purchased as a single long-living license but more based on what is needed for a limited period of time). As another example, this role will be fulfilled more and more by data center staff moving from more infrastructure-level administration tasks into higher-level tasks that encode good practices for data center management.

This base understanding is useful in identifying the main actors in the Cloud ecosystem, their requirements and their value proposition. Some organizations play the role of the service provider (typically these are organizations that own the data center or play a mediation role to service consumers), some organizations play the role of the service consumer (either end users, LOB or cloud service providers wanting to leverage third-party services) and some organizations play the role of service creators (ISVs are typically in this category as they want to leverage cloud computing as an additional delivery model for their software products).

With a strict separation of concerns the cloud architecture shown in Figure 1 addresses the requirements, of cloud solutions by providing the supporting capabilities to fulfill these requirements.

- From the service consumer's perspective, a simplified interface/API is needed with well-understood service offerings, pricing and contracts. The value proposition for the service consumer is to get much faster time to value while having to pay only for the period of time the service is used.
- From the service provider's perspective, a highly efficient service delivery and service support infrastructure and organization is needed in order to provide differentiated, well-understood, standardized and high-quality services to end



users. Service management and a dynamic infrastructure make it possible for significant economies of scale to be achieved. A self-service portal allows exposing a well-defined set of services in a highly automated fashion to a very attractive cost point.

- From the service creator's perspective, a tooling environment for modeling, assembling service elements (virtual images for example) and an effective means of managing the service lifecycle is needed.

Business requirements drive the cloud service offerings and the business support systems. The architecture must be able to support a range of service offerings, including infrastructure, platform and software/applications that are needed to support the business needs. These services offerings should be able to address both enterprises using cloud computing to supplement traditional IT as well as service providers that support multiple customers. Furthermore, with different cloud-suitable services emerging, the cloud architecture will need to provide support for workload focused offerings, including analytics, application development/test, and collaboration/e-mail services along with industry specific services. The business support systems focus on managing the business side of delivering cloud services, including managing customers, accounts, orders, subscribers, etc. Underlying these management services is the need for reporting (on usage, meeting SLAs, licenses, etc.) as well as all the capabilities for charging (including billing, invoices, settlement, etc.)

Technical requirements drive the underlying IT management patterns, including a focus on handling the top adoption factors influencing cloud services – i.e. trust, security, availability, and SLA management. The main capabilities are shown within the architecture in the operational support systems. The architecture must focus on handling the major concerns of enterprises by facilitating internal/external cloud interoperability. This requires the architecture, for example, to handle licensing and security issues to span traditional IT, private and public clouds. Additionally, the architecture must support a self service paradigm to manage clouds using a portal which requires a robust and easy to use service management solution. A portal facilitates access to the catalog of services and to manage security services. Of course, all of these services must be provided on top of a virtualized infrastructure of the underlying IT resources that are needed to provide cloud services.

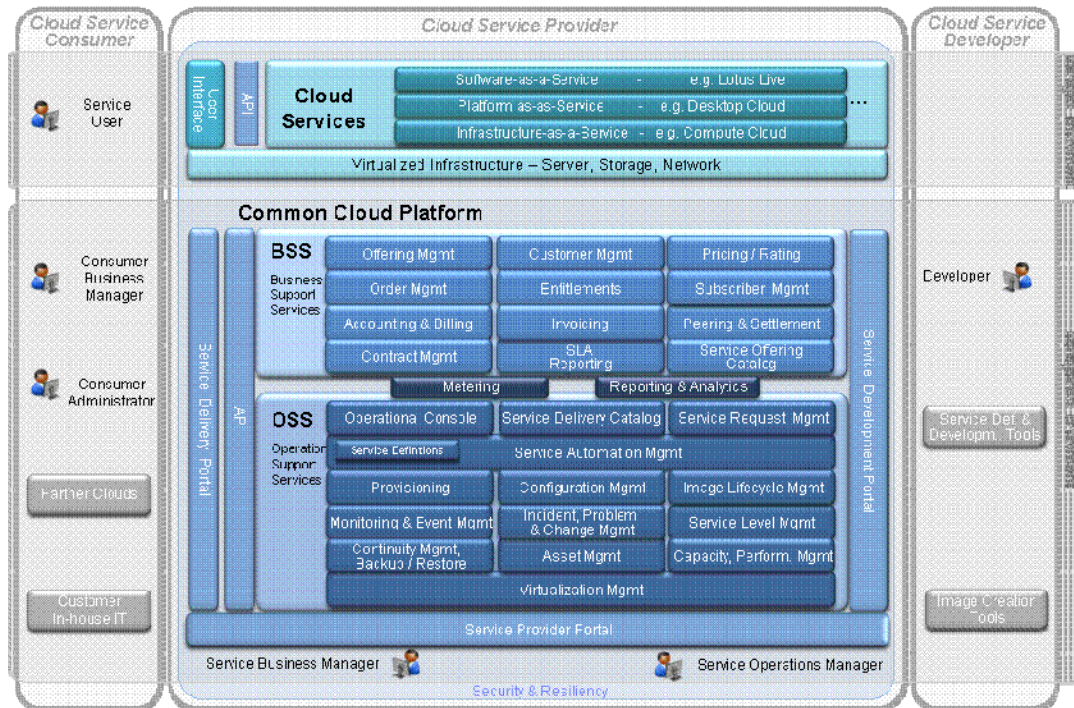


Figure 1: Cloud Reference Architecture

2 THE DESIGN APPROACH

As is evident from our discussion, cloud adoption will be a journey for any enterprise – and the architectural model will be a key component in ensuring a consistent and effective implementation through the various projects that will be used to incrementally provide cloud capabilities and allow the enterprise to derive value. Figure 2 summarizes an emerging five step program for cloud adoption. Step 1 establishes the roadmap for transforming the IT infrastructure – moving from the traditional data center approach to a centralized and consolidated infrastructure, through virtualization and automation, and finally realizing an optimized IT infrastructure capable of delivering on the promise of cloud. Step 2 establishes the architecture that will provide the capabilities needed to deliver cloud services effectively and efficiently, considering the service consumer, provider and creator. Step 3 focuses on analyzing the workloads that are feasible to move to the cloud. Step 4 focuses on determining the right mix of delivery models to deploy and use the cloud services. Finally, Step 5 lays out the implementation approach for deploying the cloud services. Note that underlying the five steps are the needs for organizational changes and governance which are critical in the transformation of the entire enterprise into the cloud mindset. Also note that the architectural model established early in Step 2 provides the basis for the rest of cloud adoption, and that the entire adoption will be an iterative process which delivers cloud capabilities incrementally. Let us discuss each step in detail.

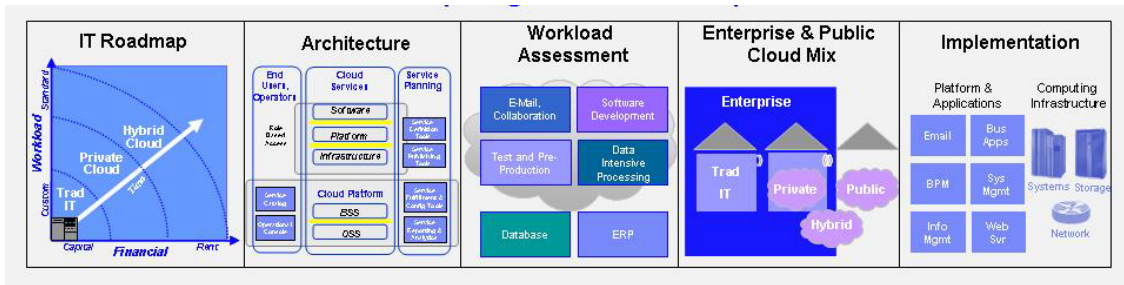
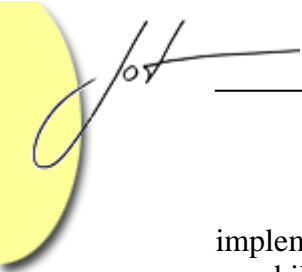


Figure 2: The Design Approach

Step 1 sets up the roadmap to transform the enterprise to delivering cloud services, and therefore establishes the plan for the cloud journey. Typically, the roadmap involves moving the business through centralization and consolidation, into virtualization and automation, and finally into an optimized cloud environment. Along this transformation, the enterprise must focus on implementing the required capabilities. In centralization and consolidation the transformation focuses on reducing infrastructure complexity and the staffing requirements on managing the infrastructure, improving business resiliency by managing fewer things better, and reducing the total cost of ownership (TCO) by improving operational costs. This is done by first establishing an enterprise data center strategy that aligns with the business needs, continuity requirements and geopolitical considerations, and then implementing the strategy including site relocation, consolidation, and new construction. The focus of consolidation is to move underutilized IT resources into larger, denser, scalable clusters, create pools of resources which can be used to service workloads, and managing and controlling these pooled resources. The next phase of the roadmap focuses on virtualization and automation. Virtualization reduces hardware costs and simplifies deployments. It accomplishes this by defining virtual resources to separate physical IT resources from its use to deliver services, establishing a single management system for virtual resources, integrating security and workload management, and controls virtual resources based on application requirements and Service Level Agreements (SLAs.) Automation focuses on dramatically reducing deployment cycles and enabling new processes/services through flexible delivery. It does this by reserving resources for applications through standardized images, automatically provisioning and de-provisioning resources based on reservations, managing workloads with integrated security and information virtualization. An optimized cloud environment will be able to sense workload requirements and move it to best-fit infrastructures while at the same time delivering qualities of service most cost effectively. It does so by optimizing workloads to maximize performance and efficiency, prioritizing workloads to attain SLAs, moving workloads to appropriate virtualized infrastructures to reduce costs, defining policies for workload management, and orchestrating workloads based on these policies.

Step 2 focuses on the cloud architecture and sets up the blueprint that will be followed throughout the cloud journey. The cloud architecture provides a guide to understanding the capabilities required for an enterprise to derive value from



implementing cloud. The reference architecture provides a comprehensive set of capabilities to ensure that cloud services can be built, deployed, accessed, delivered and managed. Each of these capabilities are supported by the appropriate standards, technologies and tools – all integrated to work together to deliver cloud computing.

Step 3 analyzes the services that the enterprise can effectively deliver in a cloud environment, and therefore establishes the scope of the cloud journey. Note that this step effectively describes the top half of the architectural model shown in Figure 2. It is important to understand the characteristics of the services in the context of a cloud delivery to determine its suitability to be delivered as a cloud service. We want to avoid services where risk and migration cost may be too high, including those that are database intensive, have complex transaction processing, packaged application (e.g. ERP) workloads, and highly regulated workloads. We want to focus on services which can be standardized and take advantage of running as a cloud service, including web infrastructure applications, collaboration infrastructure, development and test workloads, and high performance computing workloads. Finally, we should consider new workloads that have been made possible by cloud including high-volume low cost analytics, collaborative business networks, and industry focused “smart” applications

Step 4 focuses on determining how to effectively deliver the cloud services. The cloud services can be delivered on premise or off premise using multiple mechanisms – either through traditional IT services or through managed operations. The financial models can include considerations for delivering the services privately, publically or through a hybrid model. Of course, different models can be chosen based on workload characteristics and the quality of service requirements. For example, development and test workloads that do not contain sensitive data, have low security requirements, and have low qualities of service requirements can best be serviced by public cloud services.

Step 5 focuses on the implementation and deployment of the cloud service to derive business value for the enterprise. Each cloud implementation must focus on providing the capabilities for the service consumer, the service provider, and the service creator as were articulated in the cloud architecture described in Step 2. The following are key considerations in any implementation:

- The implementation should provide an easy to access, easy to use service catalog that is used by users to request services, and by administrators to publish available services.
- The implementation should hide the underlying complex infrastructure from the user and shift the focus to services provided.
- The implementation enables the ability to provide standardized and lower cost services.
- The implementation facilitates a granular level of services metering and billing.
- The implementation should ease complexity through workload standardization.



3 “ARCHITECTING” THE CLOUD SOLUTION

The cloud architecture is mainly used as a blueprint for the implementation. In this section we will show how the defined capabilities in the architecture can service the requirements posed by the typical cloud use cases associated with the main actors (service consumer, service provider and service developer) along with the typical non-functional requirements. Note that the implementation of the cloud capabilities to address these requirements requires a complete incremental and iterative software engineering methodology which will design, implement, test and deploy capabilities using the cloud architecture as a reference and guide. A description of the step-by-step approach is beyond the scope of this paper, and we will only show the mapping of capabilities required to address the requirements.

As indicated by the architecture, the following describes the key actors that must be addressed by any given cloud implementation. The service developer designs, implements, and maintains service templates that will be published in the service catalog made available to the service consumer.

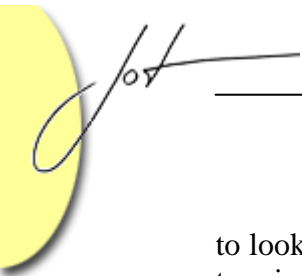
The service consumer can be further described by the following sub-roles:

- Service Consumer Business Manager who has the business/financial responsibility for all services consumed by an organization.
- Service Consumer Administrator who requests service instances and changes of service instances (typically on behalf of Consumer Business Manager) and provides access to services for the final service users.
- Service User who uses the service instances provided by service provider and requested by the Service Consumer Administrator.

The service providers offer services based on a management infrastructure. They may also develop services. Service providers can build their services by (optionally) consuming services provided by other service providers. Service providers can host services developed by other service developers (on top of their own services.) The following sub-roles provide further details about service providers:

- Service Operations Manager who manages the technical infrastructure required for providing cloud services.
- Service Business Manager who manages service catalog that list the services offered by the provider to the consumers. Accounts services consumers for services that they use.
- Service Security Manager who is responsible for ensuring that the service provider appropriately manages risks associated with development, delivery, support and use of services

We look at typical scenarios involved in any cloud implementation, and show how the architecture provides the appropriate capabilities to address these scenarios. The best way



to look at the cloud scenarios is to follow the lifecycle of a cloud service from creation to termination, and look at the how the main actors interact with the cloud service. We understand that there will be initial stages of the lifecycle involved with getting the enterprise engaged on a cloud journey, agreeing to use cloud services, and on-boarding of applications, users and administrators in the cloud environment. We leave these out of the discussion since the architecture does not have a direct impact on these initial stages.

In essence, we have three major scenarios to consider: creating a cloud service and publishing it in the service catalog, requesting/using the service, and managing the cloud environment (services, accounts, and resource pools.) The following looks at the three scenarios in more detail.

Creating a cloud service and publishing it in the service catalog involves both the cloud service developer as well as the service provider. The service developer creates service templates that will be offered to the service consumers. The type of service templates varies according to type of cloud offering. Examples of service templates include a collection of simple or composite images, storage volumes, virtual desktops, and web-conferences. Service templates are typically published in a service catalog. Underlying this service creation scenario are the steps necessary to have the appropriate resource pools to run the service, the pricing of the service (if needed), and the manner in which consumers will be on-boarded to use the services. As is clearly shown in the cloud architecture, tools are provided to the service developer to define and develop the service as well as to develop the images associated with the service. From the service provider perspective, the reference architecture provides a rich set of BSS and OSS capabilities to support the service. The BSS capabilities are associated with managing the service offering (offering management), on-boarding the customer (customer and contract management), and offering the service (service offering catalog.) The OSS capabilities include support for publishing the service (service delivery catalog), managing the images associated with the service (image lifecycle management) and setting up the resources to run the service (service automation manager.)

The next scenario in the lifecycle is requesting and using the service. The basic workflow associated with this scenario is shown in Figure 3. The service consumer requests a service from the catalog and provides pertinent information for the request including configuration choices (if available), account information (if needed) and reservation timeframes (to facilitate scheduling of the resources and reclaim them at the end of the reservation.) The service automation manager is the key component responsible for provisioning the service (by allocating the appropriate resources, installing the needed middleware and applications, and configuring it according to the service template definition) and activating it for use. The service usage is based on the service type (e.g. storing and retrieving information for a storage infrastructure services, developing and testing applications for dev/test platform services, and using email for software/application services), and includes common requirements for requesting changes to the service instance itself (e.g. more IT resources and changes in reservation dates), saving/restoring the service instance image, viewing the usage statistics and other



information of the service instances assigned to a user, and terminating the service instance.

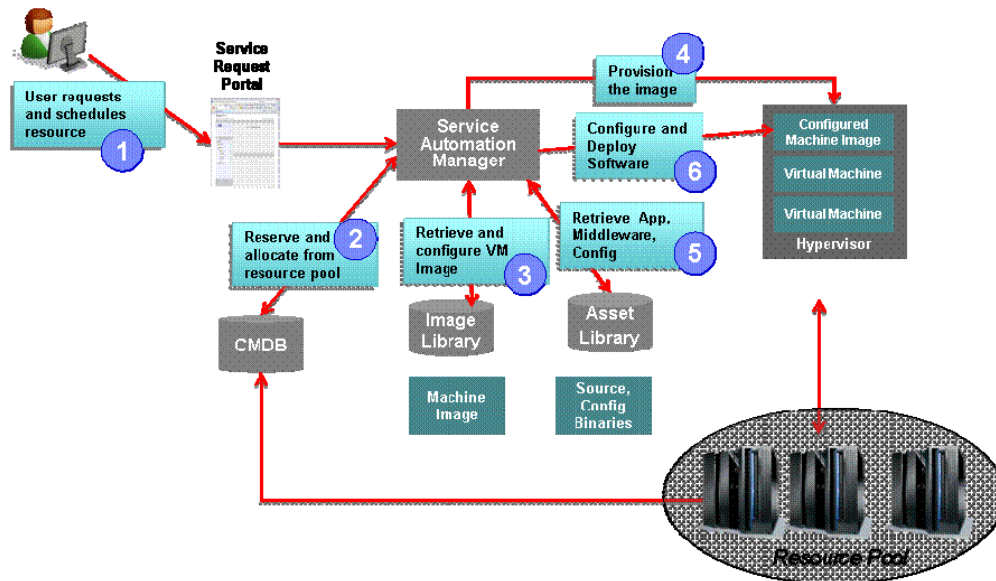


Figure 3: The Service Request Workflow

As is evident from the figure above, the key capability used from the architecture is the service automation manager that is an integral component of the OSS layer. Associated with the usage of the service are the respective portals for requesting, activating and accessing the services for both the service consumer and provider. The service automation manager uses the service delivery catalog and service request manager components to handle the requests from the service consumer. As depicted in Figure 3, it orchestrates the image lifecycle management, asset management, provisioning and virtualization management components to activate the service and support its use by the service consumer.

The final scenario for us to consider is the overall management of the cloud environment. We break this into two major types of use cases – one associated with managing the cloud services and administering the underlying resources, and the other associated with managing all aspects of the customer-related account.

For managing the cloud services and resource pools, we focus primarily on the key requirements of the service provider and its sub-roles, i.e. those that manage IT resources, virtualized environments, and cloud services. The resource administrator is interested in monitoring different types of IT resources involved in delivering cloud services, including compute, memory and storage; and the ability to manage different aspects of the IT resources, including utilization and capacity. The virtualization administrator focuses on monitoring VMs, and manages workloads associated with the cloud services as well as the performance (by increasing the IT resources allocated to the VM.) Finally, the cloud administrator monitors the entire cloud environment, and manages any

incidents and events to ensure efficient and effective delivery of the cloud services to the established SLAs. The architecture provides components to address each of these requirements – monitoring and event management to monitor different components of the cloud environments as well as handle any events generated by the cloud service; incident, problem and change management along with configuration management to address any change requests from the usage of the cloud service; performance management and service level management to monitor and manage the cloud service according to established SLAs; continuity management to ensure that the cloud service is available; and capacity management to ensure that the underlying IT resources are sufficient to handle the service requests.

The final requirement is that of managing accounts – i.e. managing all customer related data from metering usage to reporting & billing and SLA compliance, as well as managing user groups and their environment. This requires coordination between the BSS and OSS capabilities within the architecture which is captured via the metering and reporting & analytics components. The OSS components meters the usage of underlying resources by the cloud service, and makes this information available to the BSS components used for accounting and billing. The BSS components use the reporting & analytics components to gather other pertinent information from the OSS components to manage accounts – including for SLA reporting, peering & settlement, and entitlements. The user and group administration is supported by the underlying security and resiliency component in the architecture.

To summarize, it is important to have a blueprint to support your cloud implementation, and ensure that you are deriving the value that you need from your cloud solution. The cloud architecture can be leveraged effectively to ensure you have the necessary capabilities in your solution to address your requirements.

About the author



Mahesh Dodani is a software architect at IBM focusing on Cloud Computing. His primary interests are in enabling communities of practitioners to design and build solutions that address complex business needs and deliver value. He can be reached at dodani@us.ibm.com.