Realtime publishers

The Definitive Guide™ To

Cloud Computing

sponsored by

IBM®

Dan Sullivan

Copyright Statement

© 2010 Realtime Publishers. All rights reserved. This site contains materials that have been created, developed, or commissioned by, and published with the permission of, Realtime Publishers (the "Materials") and this site and any such Materials are protected by international copyright and trademark laws.

THE MATERIALS ARE PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, TITLE AND NON-INFRINGEMENT. The Materials are subject to change without notice and do not represent a commitment on the part of Realtime Publishers or its web site sponsors. In no event shall Realtime Publishers or its web site sponsors be held liable for technical or editorial errors or omissions contained in the Materials, including without limitation, for any direct, indirect, incidental, special, exemplary or consequential damages whatsoever resulting from the use of any information contained in the Materials.

The Materials (including but not limited to the text, images, audio, and/or video) may not be copied, reproduced, republished, uploaded, posted, transmitted, or distributed in any way, in whole or in part, except that one copy may be downloaded for your personal, non-commercial use on a single computer. In connection with such use, you may not modify or obscure any copyright or other proprietary notice.

The Materials may contain trademarks, services marks and logos that are the property of third parties. You are not permitted to use these trademarks, services marks or logos without prior written consent of such third parties.

Realtime Publishers and the Realtime Publishers logo are registered in the US Patent & Trademark Office. All other product or service names are the property of their respective owners.

If you have any questions about these terms, or if you would like information about licensing materials from Realtime Publishers, please contact us via e-mail at info@realtimepublishers.com.

[**Editor's Note:** This eBook was downloaded from Realtime Nexus—The Digital Library for IT Professionals. All leading technology eBooks and guides from Realtime Publishers can be found at http://nexus.realtimepublishers.com.]

Chapter 7: Roadmap to Cloud Computing: The Planning Phase

The benefits of cloud computing are well established: This model of service delivery is efficient, scales well, and meets a wide range of business needs. These benefits are maximized when business drivers, infrastructure, and policies are properly aligned to take advantage of the clouds method of delivery services. Cloud computing is not a universal panacea and some business processes are better delivered by other approaches. Not all businesses will benefit equally from cloud computing; much depends on how well they prepare for the adoption of cloud computing. The purpose of this chapter is to outline a planning process that will help maximize the benefits of cloud computing. The planning process consists of several steps:

- Assessing readiness for cloud computing
- Aligning business strategy with cloud computing services
- Preparing to manage cloud services
- Planning for centralized resources
- Committing to service level agreements (SLAs)
- Meeting compliance requirements

The chapter concludes with a pre-implementation checklist to help manage your own planning phase.

Assessing Readiness for Cloud Computing

The ancient Greek aphorism "know thy self" is surprisingly relevant to planning for cloud computing. The first step in the planning process is to assess where the organization stands with respect to

- Web application architecture
- Self-management of compute and storage services
- Standard platforms and application stacks

Each of these three areas is relevant to the delivery of cloud services. At this stage of the planning process, it is not necessary to have all three in place at ideal levels; in fact, most organizations not already supporting a cloud infrastructure will likely not have fully deployed and standardized around these three areas. This is not a problem. This is the planning process and the point of the assessment stage is to understand what resources are in place when we begin the move to cloud computing. The information gathered in this process will help to guide later planning and design efforts.

Web Application Architecture

Applications are designed using a variety of design principles that are roughly grouped into what we call application architectures. These architectures vary in terms of a number of characteristics, such as:

- Level of centralization
- Coupling of components
- Accessibility of components
- Ability to execute multiple instances
- Platform independence

We need to consider how existing applications are designed with respect to each of these to understand how well those applications are adapted to cloud infrastructure. As we will see, those with characteristics most closely aligned to Web application architectures are best suited for the cloud; but first, we will briefly describe each of these characteristics.

Levels of Centralization

An application may be centralized with all application code executing on a single machine, in a single process, and under the control of a single component. Centralized applications range from small utilities to large enterprise-scale applications. For example, a simple text editor can be realized with a single executable that runs a simple accept input-process input-generate output loop. Also in the most centralized application category, we have large, complex batch-oriented mainframe applications that have developed over years to incorporate many functions. A billing system for a telecommunications company, for example, may have millions of lines of code that, although divided into sub-modules, is largely controlled by a single control module and executes on a single machine. These applications are at one extreme of the centralization spectrum.

The middle ground of centralization is typified by client/server applications. In this application architecture, the work performed by an application is divided between servers, which perform the bulk of computing and storage operations, and client devices that are responsible for user interactions. A simple example of an application employing this approach is an order entry system consisting of a .Net user interface running on a Windows desktop and a SQL Server database. The client and the server components are fairly tightly coupled but they execute on separate devices and the components, with some effort, could be exchanged for a different form of the component. For example, the SQL Server database could be replaced with an Oracle database with little impact on the client.

Decentralized applications execute multiple processes over multiple devices. Web application architectures take advantage of decentralized applications to combine services. A typical Web application may require persistent data storage provided by a relational database, user management provided by an LDAP server, compute services provided by a Java application server, and user interaction services provided by a Web server. Decentralized applications are especially well suited for cloud architectures because services can be run on virtual servers as needed and new services can be easily added without disrupting the loose coupling between services or requiring one to provision additional dedicated hardware.

Coupling of Components

The components of an application, such as a service, module, or procedure, may be tightly coupled with other components. For example, a procedure for calculating the shipping costs of an order may be part of a larger order entry program that calls that procedure at specific points in the execution of the order entry process with a data structure specific to that program. This is an example of a tightly coupled set of components.

Loosely coupled components can execute in more autonomous ways. They may run on different servers, they may be executed on the behalf of multiple calling programs, and they exchange input and output in ways that support a broad array of calling applications. Applications built on loosely coupled components work well in cloud architectures because the number of instances can be adjusted to meet demand and the services they provide are available to other applications running in the cloud.

Accessibility of Components

Accessible components are those that are available to different services. To be accessible, a component must:

- Be programmatically discoverable so that other components can find it
- Exchange input in well-generalized formats, such as XML
- Respect authentication and authorization requirements
- Maintain reasonable response rates under varying loads

Web application architectures are built on accessible components using standards such as SOAP and WS-Security to meet some of these requirements. Others, such as the ability to maintain reasonable response rates, are met by using scalable architectures such as compute clouds.

Ability to Execute Multiple Instances

The ability to execute multiple instances might seem an odd requirement at first. After all, why couldn't one run multiple instances of an application? The answer: You couldn't run multiple instances when components are tightly coupled and exclusive use of a resource is required. A monolithic application, for example, may assume that it can lock a file of customer data for exclusive use preventing other processes from operating on that resource. If the application cannot finish processing in the time window allotted to it, the application manager could not simply start another instance of the program on a different server and finish in half the time.

Applications that are well suited to the cloud do not require that only a single instance of the program execute at any one time. Older applications may not have been designed with this characteristic in mind, but Web application architectures, built on decentralized, loosely coupled components, generally do not have these problems.

Platform Independence

Another characteristic of Web applications is that services are not required to run on a single type of platform. Services are decoupled so that requirements define how data is exchanged between those services but not how the services execute. A service that needs to retrieve information about a user could just as easily do so by calling an LDAP service running on a Linux platform as by calling Active Directory (AD) running on a Windows server.

Web application architectures are characterized by decentralized, loosely coupled components that are accessible to other service components and can scale to meet loads placed on them. This combination of characteristics is seen in the service bus model that uses message passing and service abstraction. Applications that use this approach are well suited to the cloud. Applications that do not use this model can still benefit from the management and cost benefits of using cloud services. The more decentralized and loosely coupled the application, the greater the potential benefits.

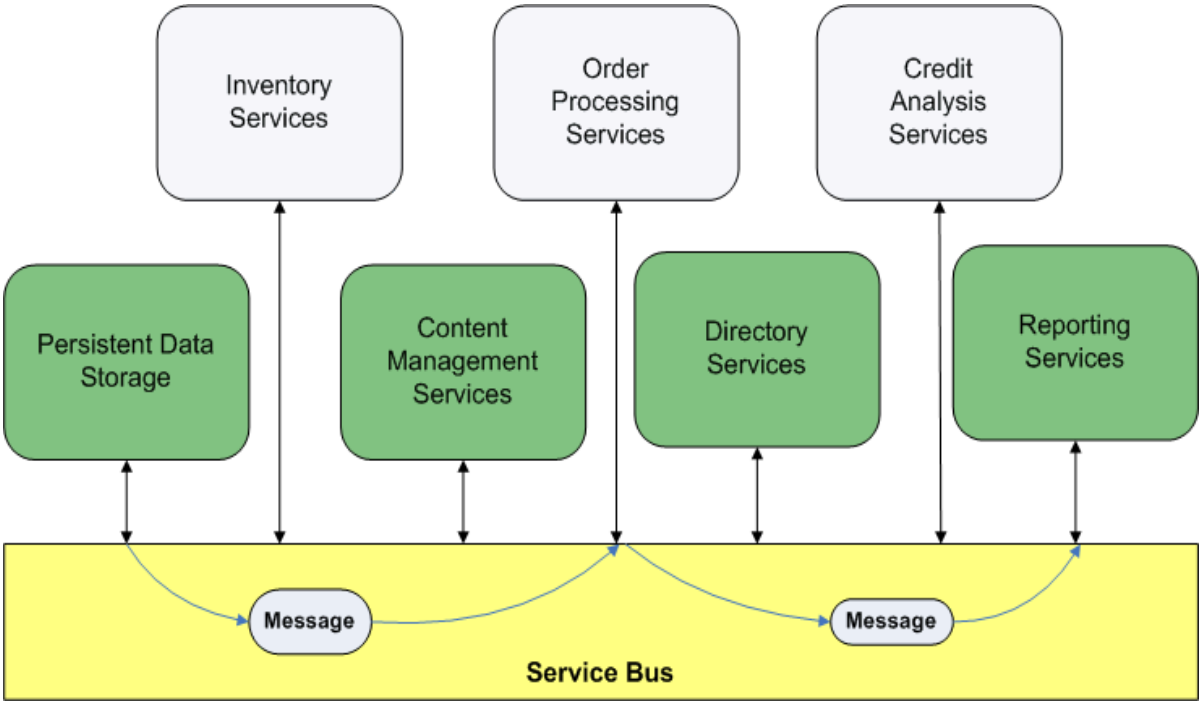

Figure 7.1: Web applications that utilize a service bus model are well suited to executing in the cloud.

From an assessment perspective, a business should try to determine how closely existing applications use a Web application architecture. Even without a formal service bus, other application architectures can exhibit the characteristics that fit well with cloud computing. For example, the common 3-tier architecture that Figure 7.2 shows has many of the characteristics previously described.

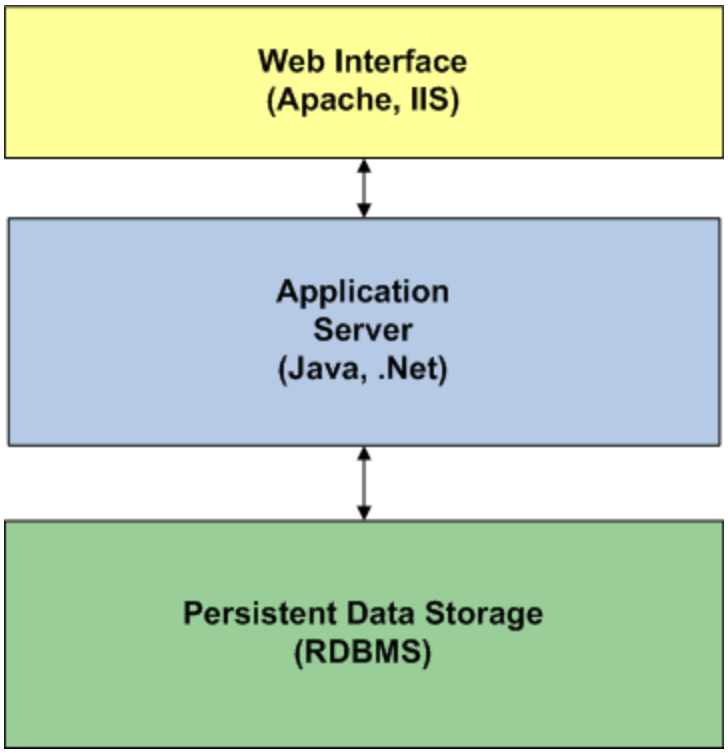

Figure 7.2: The 3-tier architecture also exhibits decentralization, loose coupling, and platform independence.

Another factor to consider when assessing readiness for cloud computing is support for self-management of resources.

Self-Management of Compute and Storage Resources

The efficient allocation of compute and storage resources requires the ability to start and stop services on demand in response to changing conditions. As we have discussed in previous chapters, one of the inefficiencies in dedicating servers to a single application is that such servers have to be configured for peak capacity and this often leads to underutilization during non-peak periods. The same problem could occur in the cloud if cloud consumers were not able to rapidly respond to changes in demand. This is true for both computing and storage resources. It is not uncommon for users of storage arrays to have to submit a ticket to IT support to have additional disk space allocated to their dedicated servers. This could take minutes to days depending on the backlog in IT support. The potential delays can lead to application managers allocating more storage than needed simply to avoid any possible risk of running out of space and not getting additional storage in time.

Ideally, application managers would be able to allocate compute and storage resources as needed. In many cases, self-management software is not in place prior to adopting cloud computing. This certainly will not prevent a business from moving to cloud computing but it will require that such a system be put in place. When evaluating compute and storage service self-management software, consider the following factors:

- Ease of use
- Management reporting for cloud consumers
- Integration with accounting and billing systems for chargeback purposes
- Adequate authentication and authorization
- Job scheduling features or integration with existing job scheduling systems
- Ability to scale to the number of cloud consumers

Another factor that will influence ease of management is the ability to standardize on platforms and application stacks.

Standard Platforms and Application Stacks

Standardizing on a limited number of operating system (OS) platforms and application stacks can ease the management of a compute/storage cloud. Many organizations may find something akin to an 80/20 rule applies to them: 80% of application needs can be met with a relatively small number of platforms and application stacks, possibly 20% of all the platforms and stacks that are currently in use in a business.

Determining Required Platforms and Application Stacks

For planning purposes, compile an inventory of applications including OSs, application servers, directory servers, Web servers, and other core components. With that inventory, one can derive a list of distinct combinations of platforms and application stacks. It is likely that many of the applications run on similar sets of OS and application stack. Those compose the "80%" side of the 80/20 rule.

For the remaining applications, assess the level of difficulty in transitioning from the existing combination of OS and application stack. For example, if many applications are running on a Red Hat version of Linux while a handful are running on SUSE versions, the effort required to migrate between those should be fairly low in most cases. An application that depends on a Windows server platform or on components that only run on Windows platforms would be significantly more difficult to port to a Red Hat platform. The goal in moving to a cloud architecture, however, is not to redesign existing applications but to leverage the benefits of the cloud.

This calls for something of a balancing act. First, we want to minimize the number of distinct application stacks we support in the cloud but we also want to maximize the number of applications that can be supported in the cloud. Adding application stacks should increase the ability to support either a significant number of general applications or targeted mission-critical applications that would benefit from running in the cloud.

Organizations that already have large portfolios of Web applications will likely find that they can address many of their requirements with a small number of different application stacks, such as:

- LAMP stack, with Linux, Apache, MySQL, and Perl/Python/PHP
- Windows stack, with .Net applications and servers
- Commonly used application servers, such as Java application servers and Java portals

Regardless of the combination of application components and OSs, there are services and policies that should be standardized across platforms in the cloud.

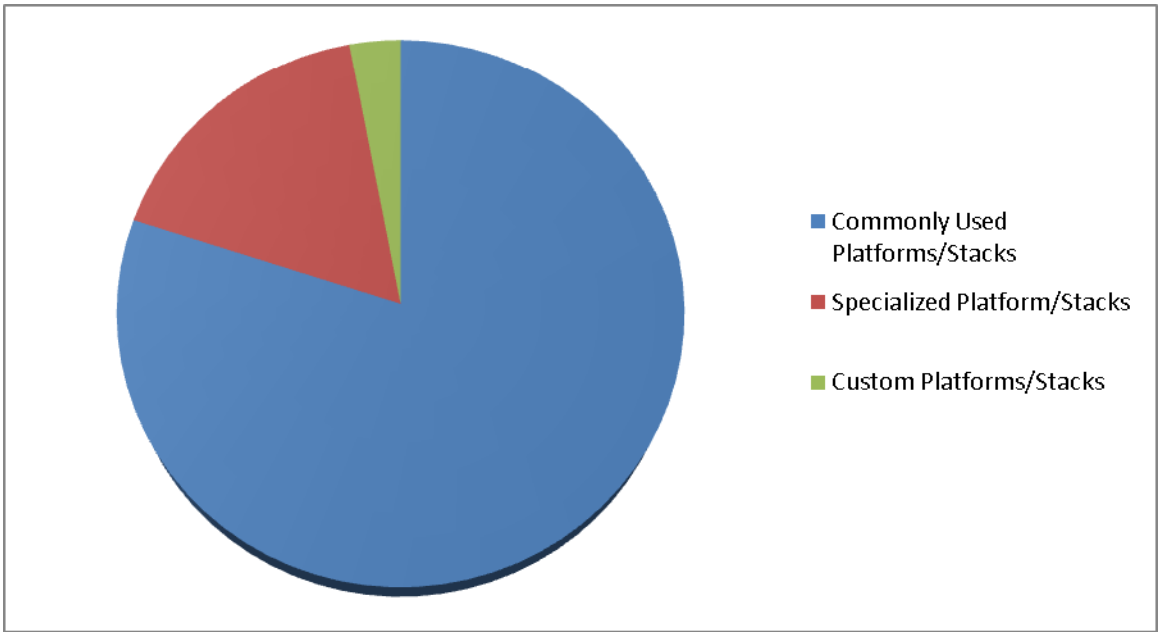

Figure 7.3: Relative distribution of platform/stack needs that can be met by a small set of commonly used stacks, specialized stacks for less common requirements, and custom platform/stacks for single, custom needs.

Required Support Services

The cloud should provide identity management services such as authentication and authorization services. These are necessary to properly administer a cloud. For example, these systems would be used to:

- Determine how users or agents are authenticated to self-service applications used to manage cloud services
- Determine limits on cloud consumers, such as the maximum number of instances a user can start at one time or the length of time a single instance can be running a single virtual machine
- Allocate charges for cloud services to the proper department or billing code

The same authentication and authorization services could be made available to applications running in the cloud, reducing the need for application-specific identity management systems.

Customization and Specialized Requirements

Another issue to consider around standardizing platforms and application stacks is the need for specialized versions of cloud-provided standards. The company may have standardized on Java or .Net for all application development but a department needs to host a third-party application developed in Ruby. Ruby is an interpreted programming language akin to Perl and Python. Ruby must be available on a server to execute a Ruby application. If this language is not part of the standard cloud offerings, the department may want to create a specialized virtual machine image to meet their needs.

There are advantages to allowing customized combinations of OSs and applications stacks. The most compelling is that cloud consumers have access to exactly what they need. There is no need to port applications to other platforms or find alternative solutions that run on standard platforms.

The disadvantage of allowing customized virtual machine instances is that they are more difficult to manage. For example, who is responsible for patching and maintaining customized virtual machine images? The creators know the components and applications best, but IT support staff may be most familiar with lower-level details, such as OS vulnerabilities. Also, if a patch were to break the application, how would it be dealt with? Will users have the knowledge and time to test patches before deploying in production? Will metadata about the contents of custom images be kept up to date? Will this task duplicate efforts already carried out by cloud providers? We are starting to see the potential for the kind of inefficiency that drives up IT costs in non-cloud environments.

Assessing readiness for moving to a cloud architecture is a critical first step in the planning process. This stage of planning requires an assessment of which applications will fit well with the cloud; those using Web application architectures, such as a service bus architecture or a multi-tiered application stack are well suited for the cloud. Once those applications are in place in the cloud, cloud consumers will want precise control over how they execute and the storage they use. Self-management services are essential to realizing the efficiencies of the cloud. Finally, during the assessment stage, one should identify the standard platforms and application stacks that are needed in the cloud. The benefits of the cloud should not be undermined by unnecessary management overhead.

The first stage of planning considered primarily technical aspects of delivering services in from a cloud. In the next stage, we consider more business-oriented aspects.

Aligning Business Strategy with Cloud Computing Services

Clouds are deployed to deliver services and services are established to meet business requirements. To ensure cloud services are deployed in a way that is aligned with business strategy, we should consider existing workloads and their corresponding value metrics.

Workload Analysis

Right now in your business there are hundreds, thousands, or even more applications executing business processes. Some of these are transaction-processing systems that provide high-volume, rapid processing of orders, inquiries, reservations, or a broad array of other narrowly focused business activities. Other applications are performing batch operations, such as generating invoices, reviewing inventory levels, or performing data quality control checks on databases. Still others are extracting data from one application, transforming the data into a format suitable for analysis, and moving it into a data warehouse. There is a wide array of different types of applications that are needed to keep an enterprise functioning.

These different types of applications have different requirements and constraints that must be considered when moving them to the cloud. For example, they might need:

- To start and finish executing within a particular time period
- To wait for another job to complete before it can begin
- To limit the functionality of some services, for example, write-locking a file to perform a backup
- To provision a significant number of servers for a short period of time for a compute-intensive operation

Any cloud will have finite resources. As part of the planning process, we need to understand what types of jobs can run in the cloud (that was addressed in the previous section) and how to run them efficiently.

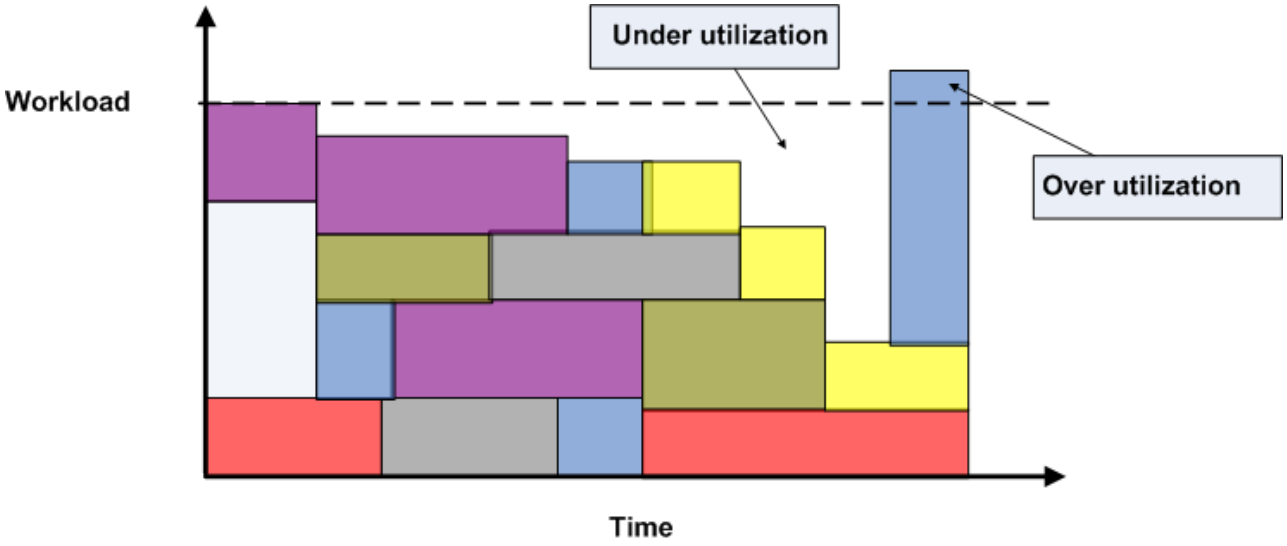

Figure 7.4: The combination of workloads running in the cloud determines overall utilization at any point in time; ideally, periods of under-utilization and over-utilization are minimized.

Cloud consumers are the ones who will decide when to start and stop jobs and how many virtual servers to provision for particular tasks, but in the planning stages, we will want to ensure there is sufficient capacity. To do so, we can look at existing workloads and take into account:

- How often jobs execute on dedicated servers
- The level of utilization of those servers
- Time constraints on when those jobs execute

Once again we are faced with a balancing act. We want to deploy sufficient cloud infrastructure to avoid periods when cloud consumers want to run more jobs than there is capacity for (over-utilization) at the same time we do not want extended periods of time when servers are idle (under-utilization). This brings us to the second aspect of business alignment: value metrics.

Value Metrics

Developing a precise and generally accepted ROI function for any IT investment is difficult at best. To assess the value of cloud computing, we can consider more targeted measures such as the value relative to hardware and software investments and relative to labor costs.

Hardware and Software Values

We will begin with hardware and software value measures by considering the constituent costs of running an application on a set of dedicated servers. They include:

- OS costs
- Application software licensing and maintenance costs
- Database management licensing and maintenance costs
- Hardware procurement and maintenance costs

The costs are relatively fixed, so it does not matter whether you run your application 24 hours a day or 1 hour a day; the hardware and software costs are the same when running that application on a dedicated server. The cost model of a cloud is different.

In a cloud model, the cost of licensing and hardware can be divided among multiple users. For example, one department might run an application for 2 hours a day, another for 6 hours a day, and a third user runs the application for 10 hours a day. Prorating the cost of licensing and maintenance over 18 hours of daily utilization lowers the cost for all three users, especially the user who only needs 2 hours of application services per day.

Labor Value

The cost of labor in the cloud model is lower than dedicated server models for a couple of reasons. First, in the cloud, there is an opportunity to standardize hardware. Large numbers of servers all built using the same, or very similar, components are easier to maintain. If a hard drive fails in a server, replace it with a spare that would work just as well in any other server. There is less overhead to manage inventory and fewer chances for errors in configuration if all servers use the same type of components.

> **Standardizing vs. Repurposing**
>
> When first deploying a cloud, you might want to repurpose hardware that had been dedicated to applications that will now run in the cloud. Some of this hardware may not match the cloud's hardware standard. Once again, we have to balance the benefits of standardizing on hardware with the cost savings of repurposing hardware. One option is to repurpose non-standard hardware but replace it with standard equipment as it fails or no longer meets functional requirements.

Second, with self-service management, cloud consumers can manage their own applications and workloads. IT support staff that had been dedicated to responding to basic server support (for example, installing software, allocating disk storage, and running backups) can now be dedicated to higher-value tasks. The cloud infrastructure will require IT support services that can be provided more efficiently in the cloud than with servers dedicated to particular applications. For example, if a vulnerability is discovered in an OS, a single administrator can patch the OS, regenerate virtual machine images, and deploy those images to the service catalog. Compare that task with the patching of hundreds of servers across the organization. By analyzing workloads and calculating initial value measure in the planning process, we are better able to align business requirements in a cost-effective way with cloud services.

Preparing to Manage Cloud Services

Up to this point in the planning process, we have considered readiness of an organization to move to a cloud architecture in terms of technical issues, such as the use of Web application architectures and standardization on platforms and application stacks. We have also examined the alignment of business strategy with cloud services in terms of workload analysis and value metrics. We now turn our attention to a few issues related to longer-term management of cloud services. These are:

- The role of private, public, and hybrid cloud services
- Planning for growth
- Long-term management issues

These issues, as we shall see, are strongly influenced by demand for cloud services.

Role of Private, Public, and Hybrid Cloud Services

There are three broad modes of delivery for cloud services: private, public, and hybrid. A private cloud is deployed and managed by an organization for its own internal use. The organization controls all aspects of cloud implementation, management, and governance. One of the most significant advantages of this approach is that data never leaves the control of its owner. This reduces the risk that an outside party will gain access to private or confidential data. Depending on the implementation and management details, private clouds may be more cost effective as well. For example, a business may have significant investment in servers that can be redeployed in the cloud, lowering the initial costs.

A public cloud is one that is managed by a third party that provides services to its customers. The primary advantage is low startup costs on the part of customers and minimal management overhead, at least with respect to basic cloud services. Businesses will still need to manage their workloads, allocate chargebacks, and so on.

Choosing between public and private cloud implementations is not an all-or-nothing proposition. Hybrid clouds, or the combination of private and public implementations to run business services, have emerged as a third alternative. Consider the economic benefits. There may be a point, however, at which the benefit of adding servers to a private cloud is not sufficient to offset the costs of adding them. For example, the distribution of workloads may entail a number of peak periods where demand exceeds the capacity of the private cloud. These peaks may be regular short periods (for example, at the end of the month when accounts are closed and data warehouses and data marts are updated and many reports are generated) or they may be more unpredictable periods of high demand.

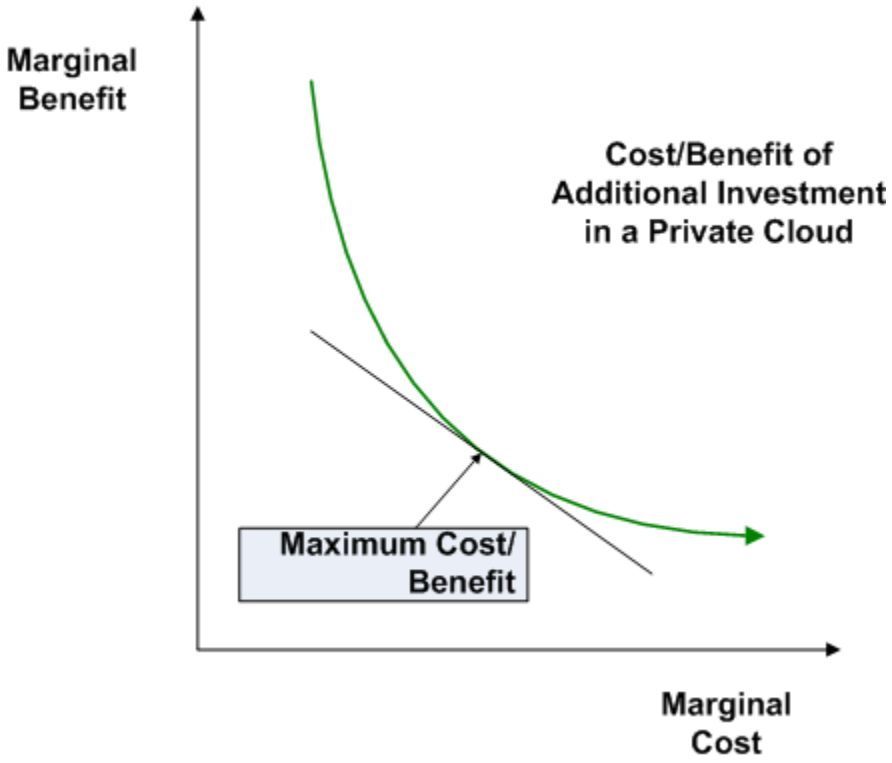

Figure 7.5: The cost of adding and maintaining additional cloud resources eventually reaches a point where the costs outweigh the benefits. At this point, a hybrid cloud approach may be the most cost-effective option.

Planning for Growth

If successful, a cloud is likely to grow both in terms of underlying infrastructure and in terms of the number of services provided by the cloud. In the case of private clouds, growth in infrastructure can occur internally by adding servers, storage, and ancillary equipment as needs demand or by adopting a hybrid cloud approach.

Growth in services will put a different kind of management burden on cloud providers. In particular, cloud providers will need to plan for:

- Expansion in the number of OSs and application stacks that may be supported
- Growing demand for custom virtual machine images to accommodate specialized requirements
- A growing base of cloud consumers with widely different needs
- Emerging categories of users, such as long-term cloud consumers who need continuously running servers, users with intermittent but regularly scheduled needs for servers, users who will take advantage of the cloud for occasional needs, or spot users who will use the cloud only during off-peak hours if the cost is lower at those times.

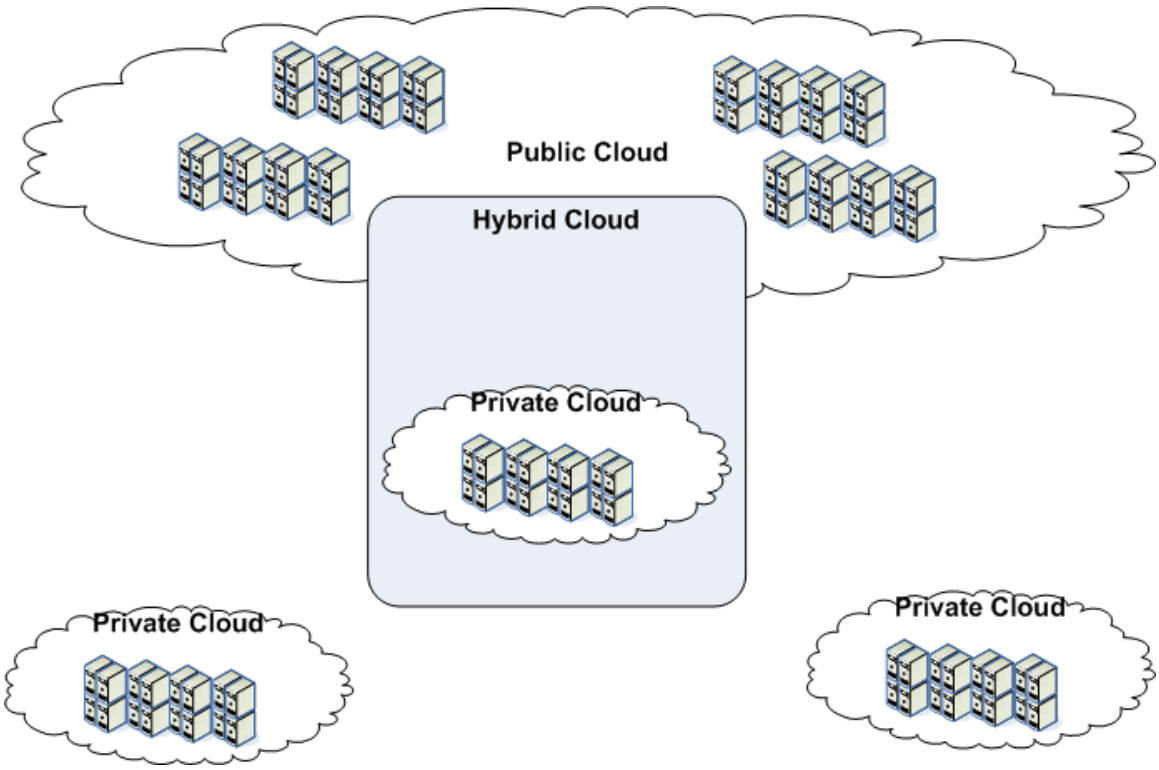

Figure 7.6: Using public cloud services in a hybrid cloud configuration during peak demand periods may be the most cost-effective way of meeting the demand for peak capacity.

These different factors will help shape management and pricing policies. A market pricing model, for instance, may be introduced to more evenly distribute the workload in cases where there are periods of high and low demand. Peak pricing could be instituted during high-demand periods and lower prices during low-demand periods. Another option is to use an auction model in which cloud consumers specify the price they are willing to pay for a resource; the cloud allocates resources to the highest bidder, then the next lower bidder, and so on until all resources are allocated.

There are many ways to manage and price services; an important point to remember is that the policies and methods used in the early days of cloud adoption may not be the best option in later stages. Following past practices because "that is the way we've always done it" is not always a recipe for success.

Long-Term Management Issues

In the planning stages for adopting a cloud, it is important to consider some of the long-term management issues that cloud providers will face. These include both service and infrastructure issues:

- Maintaining the security and integrity of virtual machine images
- Monitoring, detecting, and blocking unauthorized uses of the cloud
- Planning for high availability and disaster recovery, possibly with multiple sites for a private cloud or with the use of a hybrid cloud approach
- Managing identity, authentication, and authorization mechanisms
- Handling physical configuration of the cloud and power consumption
- Acknowledging the potential for rapid, significant rise in demands, for example with the greater use of instrumentation and data collection

These are broad issues that will continue to evolve over time. In addition to these, there are several long-term issues and responsibilities that warrant more detailed consideration.

Planning for Centralizing Resources

Cloud computing gains many of its advantages from centralizing resources, management, and governance. During the planning stage, it is important to begin formulating policies and practices that support centralization. This can come in several forms:

- Standardizing to reduce complexity
- Streamlining service management
- Virtualizing physical resources

These various forms of centralization are important individually, but they also reinforce and support the realization of each other.

Standardizing to Reduce Complexity

Standardization reduces complexity, especially in the cloud. When we use the "one server for one application" approach to delivering services, there is less need for standardization than in cloud models. That is not to stay standardization is unimportant; it is important, but the degree of standardization required to realize benefits is not as great as it is with cloud computing.

Take for example a sales department that runs a small data mart. The department had hired an analyst who had worked with open source reporting tools in the past and persuaded the department manager to use those tools as well even though the business had standardized on a commercial tool suite. The department is responsible for building and maintaining its data mart, and the group functions well with it. Centralized IT is not responsible for maintaining sales' department's system and does not object to it. (We will ignore the security implications of this decision for the moment). Now picture this application moving to the cloud.

A virtual machine image would have to be created and maintained in the service catalog of the cloud. Centralized IT management would be responsible for deploying and maintaining the image. As it is in the catalog, other users might make use of it. The user base might grow to the point that IT must spend significant time to learn the tool in order to provide support. What started as an isolated instance of using non-standard software slowly shifts to becoming an institutionalized, supported application.

Standardization is a key method of reducing complexity. The goal of standardization is to meet all functional requirements with a minimal set of computing components. Once requirements are met, adding components adds to complexity—that is, the number of interacting components that need to be maintained and adapted to function with other components—without adding to the goal of meeting requirements. In the previous example about data mart reporting, a non-standard system was used when the enterprise standard solution would have worked. The result was additional complexity with no additional benefit. Such situations should be avoided when deploying a cloud.

Streamline Service Management

One of the benefits of centralization is that by delivering services at large scales, it pays to invest in optimizing those services. A fast food chain that serves millions of sandwiches a year will optimize every aspect of the production, preparation, and delivery of those products. Similarly, the fact that hundreds or thousands of users will repeatedly invoke the same standardized set of services demands attention to streamlining and optimizing the delivery and management of those services.

In order to streamline service management, we need applications in place that reduce the manual labor and complexity of workflows required to implement management processes. In particular, service management should include:

- Support for discovering services provided in the cloud through detailed and up-to-date metadata about services
- Virtual machine images that are designed to support services, such as report generation, and not just OSs and application stacks, such as Linux with a statistical analysis package installed
- Management reporting that allows cloud consumers to track and optimize their own use of cloud resources

- Ability to provide timely support for cloud consumers in cases where there are problems executing jobs in the cloud
- Utilization analysis reports to give those responsible for managing cloud services the information they need to detect trends and analyze varying patterns of resource utilization

One of the factors that supports the ability to streamline service management is the ability to virtualize cloud infrastructure.

Virtualizing Physical Resources

The final aspect of centralizing resources we will consider is the need to virtualize physical resources. As we have encountered repeatedly within our discussion of cloud computing, the ability to virtualize computing and storage services are at the foundation of the efficiencies provided by the cloud model. The key physical resources that should be virtualized are servers and storage.

Setting up a set of virtual machines on a single server is straightforward: install a hypervisor and create virtual machine instances based on OS(s) of choice. Scaling virtualization to a large number of servers requires management software that can manage multiple hypervisor clients from a single console.

Storage services also need to be virtualized so that they appear to cloud consumers to be a single storage device. Virtual machine instances in the cloud, for example, should be able to address storage space on the cloud SAN(s) without having to manage implementation details. Ideally, the same management console that is used to control servers in the cloud will support management and administration of storage resources.

Computing and storage clouds hide many of the implementation details that go into building and maintaining a large IT infrastructure. By standardizing services, streamlining service management, and virtualizing physical resources, cloud providers enable the technical resources needed by users to leverage cloud services. Those same users, however, also require attention to business considerations.

Committing to SLAs

Business managers may look at cloud services and find the lower costs, greater control, and potential for scaling business processes compelling reasons to use cloud services. These reasons are often not enough, though. It is not sufficient for a cloud to work well today; it needs to work well for as long as users need it. This is why we have SLAs. SLAs are standard in IT, and it is no surprise that they are used with cloud services. Rather than focus just on the availability of a specific application, cloud SLAs may be more general and apply to capacity commitments, network infrastructure, storage infrastructure, and availability and recovery management. These SLAs are closely coupled to the infrastructure of the cloud, but the primary concern is on the business commitments cloud providers make to their customers.

Capacity Commitments

A capacity commitment in an SLA outlines the number and types of server capacity that will be available for use when the cloud consumer attempts to use them. Several factors should be considered when making capacity commitments:

- The total infrastructure planned for a private cloud
- The ability to acquire additional resources (compute and storage) as needed through a hybrid cloud
- Changes in pricing models if hybrid resources are used
- A commitment to the percent of time the capacity will be available
- Length of time the capacity will be available without interruption once the capacity is provisioned

The workload analysis performed earlier in the planning process can help to understand the capacity commitments a cloud provider can make given a particular number of servers and storage capacity.

Network Infrastructure

Network service commitments are especially important when there are high levels of data exchange in and out of the cloud. Service commitments will be limited by the network capacity of Internet service providers (ISPs) and the ability to distribute networking load across multiple ISPs. Cloud service providers are limited by the service level commitment they receive from their ISPs; however, by combining network services from multiple providers, a cloud provider can improve total throughput and availability.

Storage Infrastructure

Storage SLAs take into account several factors:

- Amount of storage available for use
- Backup services, if any
- Availability commitments, including percent of time storage services will be available
- Throughput commitments

When considering the amount of storage available for use, take into account the need for redundant storage to improve performance and availability. These can significantly reduce the total amount of storage available for direct use by cloud consumers.

Availability and Recovery Management

Another popular topic for SLAs is recovery management. The redundancy of servers in the cloud ensures that the failure of a single server in the cloud will not disrupt an operation. The service can be started again on another server. From a service level perspective, cloud providers may be able to commit to high levels of availability in terms of having servers available to run applications. One must account for the fact, though, that when a server fails and another is started in its place, there may be data loss depending on how the application is written. If the application writes state information to cloud storage, another instance of the application can recover from the last point at which state information was written to the disk. If the application depends on maintaining state information in memory, the recovery point would be earlier. A final set of issues that falls under the penumbra of business drivers is compliance requirements.

Compliance Requirements and Cloud Services

Compliance requirements tend to focus on preserving the integrity of data, especially financial data, and protecting the privacy of confidential information. One of the greatest impediments to adopting public cloud computing is concern about protecting the integrity and confidentiality of data once it leaves the corporate-controlled network. Private clouds retain data within corporate firewalls where it will be subject to internal controls. The assumption behind this reasoning is that governance procedures that protect data in non-cloud infrastructure are sufficient to protect the same data in the cloud. This may be true for the most part, but the cloud introduces additional factors that should be considered:

- Applications running in a virtual machine might write data to local disks. When the virtual machine shuts down, all data written by it should be overwritten.
- Authorizations assigned to users for non-cloud resources should be respected in the cloud. For example, if data moves from a dedicated file server to cloud storage, the same restrictions on access should apply.
- Practices employed as part of compliance efforts, such as routine vulnerability scanning, will have to be adapted to scan machine images in the service catalog rather than just instances running at a particular point in time on a given set of servers

Reporting is another essential part of compliance. It is not sufficient to be in compliance; one must often be able to demonstrate one is in compliance. Again, existing procedures might need to be modified to accommodate reporting on cloud procedures that support compliance. For example, each time a virtual machine instance is shut down, a record may be logged indicating local data has been overwritten to prevent the next user from scanning local storage for residual data.

Summary

Planning for cloud services is a multifaceted process that begins with assessing readiness for the cloud and aligning business strategy with cloud computing services. It also requires preparation for managing cloud services and planning for centralized resources. In addition, it entails a number of business-oriented concerns, such as SLAs and support for compliance efforts. To facilitate the planning process, a pre-implementation checklist is provided that summarizes the key points of this chapter.

Pre-Implementation Checklist	
Assessing Readiness for Cloud Computing	Determine whether applications are designed to use a Web application architecture, service bus architecture, or n-tier architecture
	Assess ability to provide for self-service management of computing and storage services
	Standardize on platforms and application stacks
Aligning Business Strategy with Cloud Computing Services	Analyze workloads
	Determine value metrics with respect to labor, hardware, and software
Preparing to Manage Cloud Service	Understand the roles of private, public, and hybrid clouds and their utility for business requirements
	Plan for growth in demands for services
	Assess long-term management issues
Committing to SLAs	Perform capacity planning with respect to service level commitments
	Analyze capacity of network infrastructure
	Analyze capacity of storage infrastructure
	Formulate reasonable commitments with respect to availability and recovery management
Meeting Compliance Requirements	Determine security requirements for preserving the integrity and confidentiality of data
	Adapt reporting requirements to address compliance implementation issues introduced by the cloud

Download Additional eBooks from Realtime Nexus!

Realtime Nexus—The Digital Library provides world-class expert resources that IT professionals depend on to learn about the newest technologies. If you found this eBook to be informative, we encourage you to download more of our industry-leading technology eBooks and video guides at Realtime Nexus. Please visit http://nexus.realtimepublishers.com.