O'REILLY®

"Finally, an industry insider provides actionable guidance on how to begin your cloud journey..."
—Wes Hogentogler, Director, Pure Integration

Compliments of NGINX

The Enterprise CLOUD

Best Practices for Transforming Legacy IT

James Bond

The Enterprise Cloud

Best Practices for Transforming Legacy IT

James Bond

This Excerpt contains Chapters 3 and 4 of the book *The Enterprise Cloud*. The full book is available on oreilly.com and through other retailers.

Beijing • Boston • Farnham • Sebastopol • Tokyo O'REILLY®

The Enterprise Cloud
by James Bond

Printed in the United States of America.

Published by O'Reilly Media, Inc., 1005 Gravenstein Highway North, Sebastopol, CA 95472.

O'Reilly books may be purchased for educational, business, or sales promotional use. Online editions are also available for most titles (*http://safaribooksonline.com*). For more information, contact our corporate/institutional sales department: 800-998-9938 or corporate@oreilly.com.

Editor: Brian Anderson
Production Editor: Shiny Kalapurakkel
Copyeditor: Bob Russell, Octal Publishing, Inc.
Proofreader: Jasmine Kwityn
Indexer: Wendy Catalano
Interior Designer: David Futato
Cover Designer: Karen Montgomery
Illustrator: Rebecca Demarest

May 2015: First Edition

Revision History for the First Edition

2015-05-15: First Release

See *http://oreilly.com/catalog/errata.csp?isbn=9781491907627* for release details.

978-1-491-90762-7

[LSI]

Contents

Foreword

During the past few years, we've seen innovative startups like Airbnb, Netflix, and Uber shoot up from small challengers to category leaders. These companies have built amazing products that have allowed them to quickly capture tens of millions of users, an achievement that only a decade ago we would have expected only from large, established corporations with huge budgets.

How did they reach these heights? Companies like Netflix were among the first to take advantage of a new and better way to develop and deliver apps: adopting DevOps processes and deploying in the cloud. Today, Netflix deploys new features within minutes, while managing a portfolio of over 100 services running on tens of thousands of servers that serve over one billion hours of content each month.

More and more enterprises are adopting the "Cloud-plus-DevOps" approach to achieve business goals and stay competitive. The transition from internal enterprise IT to the cloud promises to be the most significant change in the history of corporate computing.

Migrating enterprise applications to the cloud is difficult, however. There are many ways to deploy applications in the cloud, and each requires a certain set of tools and knowledge. At NGINX, we are proud of our role in helping enterprises move their applications to the cloud by providing an easy to deploy, software-based application delivery platform that solves the challenges of performance, reliability, scalability, security, and monitoring of applications.

So far, enterprises have lacked a prescriptive industry specification to guide them as they move to the cloud. By reading this ebook, you'll learn from industry leader James Bond about planning your long-term cloud strategy, along with many tips, insider insights, and real-world lessons about planning, design, operations, security, and application transformation as you migrate to the cloud. We hope you enjoy this ebook.

—Patrick Nommensen, NGINX, Inc.

Deploying Your Cloud

Key topics in this chapter:

- The consume-versus-build decision
- Building your own cloud—lessons learned, including architecture examples and guidance
- Managing scope, releases, and customer expectations
- Redundancy, continuity, and disaster recovery
- Using existing operational staff during deployment
- Deployment best practices

Deciding Whether to Consume or Build

A critical decision for any organization planning to use or build a cloud is whether to consume services from an existing cloud service provider or to build your own cloud. Every customer is unique in their goals and requirements as well as their existing legacy datacenter environment, so the consume-versus-build decision is not always easy to make. Cloud systems integrators and leading cloud providers have learned that there is often customer confusion with regard to the terms *public cloud*, *virtual private cloud*, *managed cloud*, *private cloud*, and *hybrid cloud*.

Figure 1-1 presents a simplified decision tree (from the perspective of you as the cloud customer or client) that explains different consume-versus-build options and how they map to public, private, and virtual private clouds. For definitions and comparisons of each cloud deployment model, refer to ???.

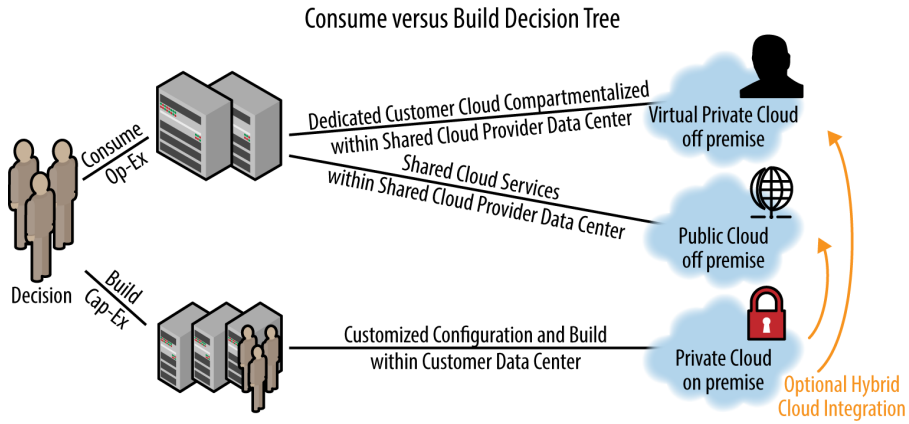

Figure 1-1. The consume-versus-build decision tree

To better understand the consume-versus-build options that are presented in the following sections, read ??? in ???.

CONSUMPTION

Consumption of cloud services refers to an organization purchasing services from a cloud service provider, normally a public cloud. In this consumption model, there are little or no up-front capital expenses; customers incur service fees on a daily, weekly, monthly, or yearly basis, which cover subscribed services and resource usage. This model requires no on-premises computing infrastructure be installed at the customer's facility or added to its network.

Also understand that a consumption model also can apply to a virtual private cloud or managed cloud that is fully hosted at a cloud provider's facility. This is just like a public cloud subscription model but with some level of customization and usually a private network compartment. Cloud providers might require some level of minimum quantity or term commitment and possibly some initial capital investment to configure this virtual private or managed cloud on behalf of the customer organization.

BUILD

Building a cloud service is usually for a private cloud deployment that can be located in a customer's datacenter or chosen third-party datacenter. In this private cloud deployment model, the customer would normally take on the burden of most or all of the capital expenses to deploy and manage the cloud service. The customer organization can chose to use a systems integrator that has expertise in

deploying private clouds, or the organization can design and procure all of the hardware and software components to build its own cloud. Experience demonstrates that hiring a systems integrator that specializes in private cloud deployment results in a faster deployment, lower risk, and a more feature-rich cloud environment—more important, this allows your organization to focus on your core business and customers rather than trying to also become a cloud integration and deployment expert.

There are numerous business decisions to be made in the planning process when building your own cloud for internal, peer, or suborganizational consumption. You will need to determine who your target consumers or organizations are, which cloud deployment model to start with, how you will sell (or chargeback) to end consumers or users, and how you will govern and support your customers. The technical decisions, size of your initial cloud infrastructure and your cloud management system will also vary depending on your business decisions.

CLOUD DEPLOYMENT MODELS

The first thing to decide is what cloud deployment type best fits your target end consumers (your own organization or customers/consumers of your cloud service for which you might host IT services). These are the choices you'll need to decide among (for complete descriptions and comparison of each cloud model, refer to ???).

Public cloud

If you truly want to become a public cloud provider, you need to build your infrastructure, offerings, and pricing with the goal of establishing a single set of products standardized across all of your customers. These cloud services would typically be available via the Internet—hence the term public—for a wide number and array of target customers. You would normally not offer customized or professional services; instead, you would focus on automation and as little human intervention in the management of your system as possible, keeping your costs low and putting you in a competitive position.

Virtual private cloud

Another offering gaining popularity is called the virtual private cloud, which is essentially a public cloud provider offering a unique (i.e. private) compartment and subnetwork environment per customer or tenant. This smaller private subcloud—or cloud within the larger cloud—can have

higher security and some customization, but not to the level possible with a pure private cloud.

Private cloud

If you plan to offer your internal enterprise organization or customers a personalized or custom cloud solution, a private cloud deployment model is likely your best choice. You can deploy and host the private cloud within any customer or third-party datacenter. One of the most important technologies that must be evaluated, procured, and installed is a cloud management platform that will provide a customer ordering and subscription management portal, automated provisioning of Anything as a Service (XaaS), billing, and reporting.

Community cloud

Community clouds are essentially a variation of private clouds. They are normally designed and built to the unique needs of a group of organizations that want to share cloud infrastructure and applications. Some portions of the organization host and manage some cloud services, whereas other portions of the organization host a different array of services. There are countless possible variations with respect to who hosts what, who manages services, how procurement and funding is handled, and how the end users are managed. The infrastructure, network, and applications deployed for a community cloud will depend upon the customer requirements.

Hybrid cloud

Most organizations using a private cloud are likely to evolve into a hybrid model. As soon as you connect one cloud to another, particularly when you have a mix of public and private cloud services, you have, by definition, a hybrid cloud. Hybrid clouds connect multiple types of clouds and potentially multiple cloud service providers; connecting to a legacy on-premises enterprise cloud is also part of a hybrid cloud. You can deploy a hybrid cloud management system to coordinate all automation, provisioning, reporting, and billing across all connected cloud service providers. Hybrid cloud management systems, sometimes call *cloud broker platforms*, are available from several major systems integrators and cloud software vendors for deployment within an enterprise datacenter or private cloud. After you deploy one, you can configure these hybrid cloud management systems to integrate with one or more external cloud providers or legacy datacenter IT systems.

Cloud Infrastructure

There are significant consume-versus-build decisions to be made when you're determining the design and deployment of the cloud infrastructure. The cloud infrastructure includes everything from the physical datacenters to the network, servers, storage, and applications.

The cost of building and operating the datacenter is extremely expensive, and sometimes not within the expertise of non-technology-oriented organizations. Professional cloud service providers, systems integrators, or large organizations with significant IT skills are better suited to managing an enterprise private cloud infrastructure.

DATACENTERS

Most cloud service providers will have at least two geographically diverse datacenters; thus, loss of either does not interrupt all services. Cloud providers (or organizations building their own enterprise private cloud) can either build their own datacenters or lease space within existing datacenters. Within each datacenter is a significant amount of cooling and power systems to accommodate housing thousands of servers, storage devices, and network equipment.

Modern datacenters have redundancy built in to everything so that any failures in a single component will not harm the equipment hosted within. Redundant power systems, battery backup, and generators are deployed to maintain power in the event of an outage. Datacenters have a certain amount of diesel fuel housed in outdoor or underground tanks to run the generators for some period of time (24 to 48 hours is typical) with multiple vendors prearranged to provide additional fuel if generator power is needed for a longer period of time.

Similar to the redundancy of the power systems, the cooling systems within a datacenter are also redundant. Given the vast number of servers and other equipment running within the facility, maintaining the interior at an ideal temperature requires a significant amount of HVAC equipment. This is of paramount concern because prolonged high temperatures will harm the network and computer infrastructure.

The power required by a datacenter is so significant that often, a datacenter can become "full" because of the lack of available power even if there is available physical space within the building. This problem is exacerbated by high-density servers that can fit into a smaller space but still require significant power.

Physical security is also a key component. Datacenters are often housed in unmarked buildings, with a significant amount of cameras, security guards, bio-

metric identity systems, as well as interior cages, racks, and locks separating sections of the floor plan. Datacenters use these tools to ensure that unauthorized personnel cannot access the computer systems, which prevents tampering, unscheduled outages, and theft.

Figure 1-2 presents a simplified view of a typical datacenter. The three lightly shaded devices with the black tops, shown on the back and front walls of the floor plan, represent the cooling systems. The dark-gray devices are the power distribution systems. The medium-shaded racks contain servers, and the remaining four components are storage and data backup systems. Notice that I don't show any network or power cables: those are often run in hanging trays near the ceiling, above all the equipment. These depictions are for explanatory purposes only; actual equipment within a datacenter varies greatly in size and placement on the floor.

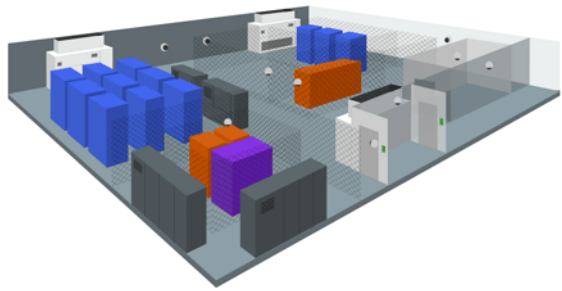

Figure 1-2. A simplified view of a datacenter's interior components

Datacenters sometimes contain pods (cargo type containers preconfigured with servers, network, and storage infrastructure), caged areas, or rooms, each with similar equipment to that shown in Figure 1-2. Cloud providers or customers often lease out entire pods until they are full and then begin filling additional pods as necessary.

NETWORK INFRASTRUCTURE

The "vascular" system of a cloud service provider is the network infrastructure. The network begins at the *edge*, which is where the Internet communication circuits connect to the internal network within the datacenter. Network routers and firewalls are typically used to separate, route, and filter traffic to and from the Internet and the internal network. The network infrastructure consists of everything from the edge and firewall to all of the datacenter core routers and switches, and finally to each top-of-rack (ToR) switch.

INTERNET SERVICES

For customers to use the cloud services, the cloud provider needs to implement a fairly large and expandable connection to the Internet. This connection often includes purchasing bandwidth from multiple Internet providers for load balancing and redundancy; as a cloud service provider, you cannot afford to have Internet connectivity lost. Because the amount of customers and traffic are likely going to rise over time, ensure that the agreement with your Internet providers allows for increasing bandwidth dynamically or upon request.

INTERNAL NETWORK

Within the datacenter, the cloud provider's network typically begins with routers and firewalls at the edge of the network connected to the Internet communication circuits. Inside the firewalls are additional routers and core network switching equipment with lower-level access switches cascading throughout the datacenter. The manufacturer or brand of equipment deployed varies based on the cloud provider's preference or skillset of the network engineers. Like computers, network equipment is often replaced every three, five, or seven years to keep everything under warranty and modern enough to keep up with increasing traffic, features, and security management.

Today's internal networks are not only for traditional Internet Protocol (IP) communications between servers, applications, and the Internet; there are numerous other network protocols that might need to be supported, and various other forms of networks such as *iSCSI* and *Fibre Channel* that are common to storage area networks (SANs). Cloud providers might decide to use networking equipment that handles IP, SAN, and other forms of networking communications within the same physical network switches. This is called *converged networking* or *multiprotocol/fabric switches*.

Because networking technologies, protocols, and speeds continue to evolve, it is recommended that you select a manufacturer that continuously provides new and improved firmware and software. Newer versions of firmware or software include bug fixes, newer features, and possibly newer network protocols. Some networking equipment is also very modular, adding small modules or *blades* into a shared chassis, with each module adding more network capacity, or handling special functions such as routing, firewalls, or security management.

The diagram shown in Figure 1-3 is a simplified view of a sample network infrastructure. The top of the network begins where the Internet connects to redundant *edge routers* that then connect to multiple lower-layer distribution and

access switches cascaded below the core. The final end points are the servers, storage devices, or other computing devices. What is not shown, but is typical, are multiple network circuits connected to each end-point server to provide load-balanced, redundant, or *out-of-band network* paths. These out-of-band network paths are still technically part of the network, but are dedicated communications paths used for data backups or management purposes. By keeping this network traffic off of the production network, data backup and management traffic never slows down the production network.

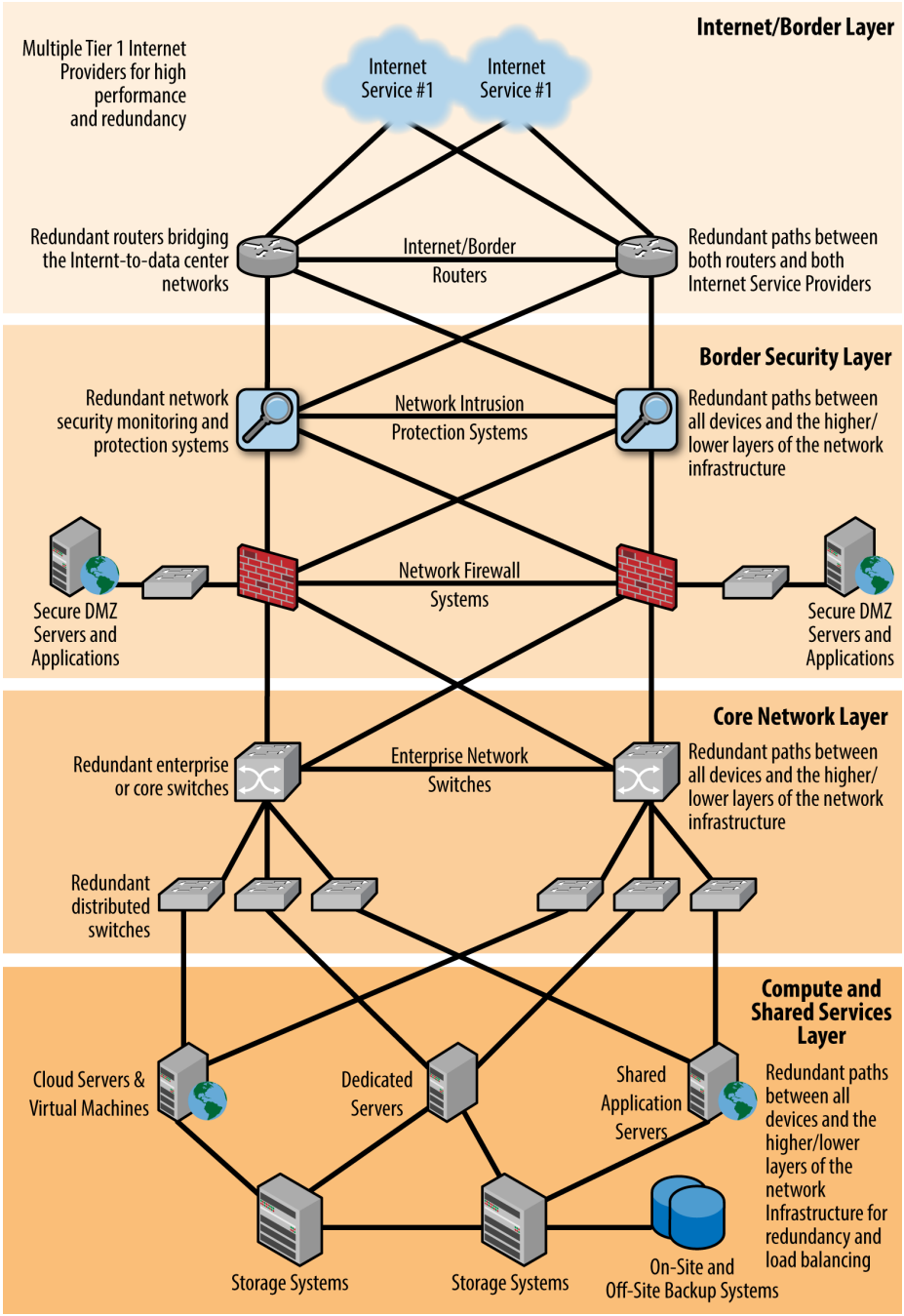

Figure 1-3. Network infrastructure layers

COMPUTE INFRASTRUCTURE

The compute infrastructure is where the physical servers are deployed within a datacenter. Not long ago, datacenters were filled with traditional "tower" servers; however, this has since shifted to a higher-density rack-mounted form factor. To fit even more servers and compute power into precious rack space, blade servers are now the norm; a single blade cabinet within a rack can hold a dozen or more plug-in blade-server modules. With the rack capable of holding three or four of these cabinets, you can achieve more server compute power in a single rack than ever before. However, the amount of power and cooling available per rack is often the limiting factor, even if the rack still has physical space for more servers.

Modern rack-mount and blade servers can each house 10 or more physical processors, each with multiple processor cores for a total of 40 or more cores. Add to this the ability to house up to a terabyte of memory in the higher-end servers, and you have as much compute power in one blade server as you had in an entire rack back in 2009.

Here is where cloud computing and virtualization comes into play. There is so much processor power and memory in today's modern servers, that most applications cannot utilize all of the capabilities efficiently. By installing a hypervisor virtualization software system, you can now host dozens of virtual machines (VMs) within each physical server. You can size each VM to meet the needs of each application, rather than having a lot of excess compute power going unused if you were to have just one application per physical server. Yes, you could simply purchase less powerful physical servers to better match each application, but remember that datacenter space, power, and cooling come at a premium cost; it makes more sense to pack as much power into each server, and thus into each rack, as possible. When purchasing and scaling your server farms, it is now common to measure server capacity based on the number of physical blades multiplied by the number of VMs that each blade can host—all within a single equipment rack.

Figure 1-4 shows a simplified view of a small two-datacenter cloud environment. This example shows both physical and virtual servers along with the cloud management platform, identity authentication system, and backup and recovery systems spanning both datacenters. In this configuration, both physical servers and virtual servers are shown, all connecting to a shared SAN. All SAN storage in the primary datacenter is replicated across to the secondary datacenter to facilitate disaster recovery and failover should services at the primary datacenter fail. Notice that the firewall located in the center indicating a secure network connec-

tion between datacenters that allows traffic to be redirected. This also facilitates failover of both cloud management nodes and guest/customer VMs if necessary. In the event of a total outage at the primary datacenter, the secondary datacenter could assume all cloud services.

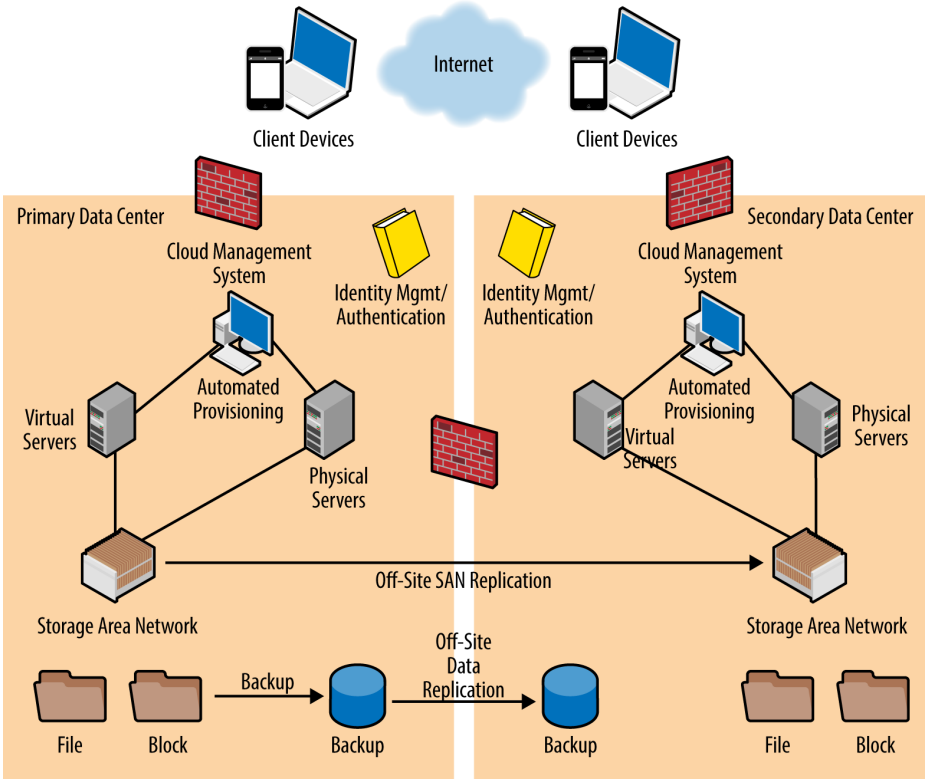

Figure 1-4. Typical network and server infrastructure (logical depiction)

Here are some factors you should consider when selecting and deploying the compute infrastructure:

Server hardware

A typical cloud infrastructure would start with one or more server-blade chassis populated with multiple high-density blade servers (see Figure 1-5). You should consider standardizing on a single vendor so that your support staff can focus on one skillset, one brand of spare parts, and one server management system. Most servers in this category have built-in manage-

ment capabilities to remotely monitor and configure firmware, BIOS, and many other settings. You can also remotely reboot and receive alerts on problems from these built-in management capabilities. Forwarding these alerts, or combining the manufacturer's management system with a larger enterprise operations system, is ideal for completely integrated management. In a cloud environment, servers and blade chassis that have advanced virtualization and software-defined mappings between chassis, blades, storage, and networking present a significant advantage—so a "cloud enabled" server farm is not just marketing hype, but a real set of technologies that cloud providers can take advantage of.

CPU and memory

The performance and quantity of the processors in a server varies by manufacturer and server model. If you were trying to host the maximum amount of VMs per physical server or blade, you would want to select the maximum amount of processor power that you can get within your budget. Often, purchasing "last year's" newest and best processor will save you significant money, compared to buying the leading-edge processor, just released, at a premium markup. The amount of memory you order within each physical server will depend on the amount of processors you order. Overall, try to match processor to memory to see how many VMs you can host on each physical server. Popular hypervisor software vendors—Microsoft, VMware, KVM, Citrix, and Parallels—all have free calculators to help size your servers appropriately.

Internal versus external hard drives

Most rack or blade servers have the ability to hold one or more internal hard drives; the question is less about the size of these hard drives, but if you really want any at all inside each server. I highly recommend not installing local hard drives; instead, use shared storage devices that are connected to the blade server chassis via a SAN technology such as Fibre Channel, iSCSI, or Fibre Channel over Ethernet (FCOE), as depicted in the logical view in Figure 1-4. (A physical view of the SAN storage system is shown in Figure 1-5.) The servers boot their operating system (OS) from a logical unit numbers (LUN) on the SAN, rather than from local hard drives. The benefit is that you can install a new server or replace a blade server with another one for maintenance or repair purposes, and the new server boots up using the same LUN volume on the SAN. Remember, in a vast server farm, you

will be adding or replacing servers regularly; you don't want to take on the burden of managing the files on every individual hard drive on every physical server. Also, holding all of the OS, applications, and data on the SAN allows for much faster and centralized backup and recovery. Best of all, the performance of the SAN is several times better than that of any local hard drive; in an enterprise or cloud datacenter, you absolutely need the performance that a SAN provides.

It should be noted that SANs are significantly more expensive than direct-attached storage (DAS) within each physical server; however, the performance, scalability, reliability, and flexibility of configuration usually outweigh the cost considerations. This is especially true in a cloud environment in which virtualization of everything (servers, storage, networking) is critical. There are storage systems that take advantage of inexpensive DAS and software installed across numerous low-cost servers to form a virtual storage array. This approach uses a large quantity of slower-speed storage devices as an alternative to a high-speed SAN. There are too many features, costs and benefits, performance, and operational considerations between storage approaches to cover in this book.

Server redundancy

Just as with the network infrastructure described earlier in this chapter, redundancy also applies to servers:

Power and cooling

Each server you implement should have multiple power supplies to keep it running even if one fails. In a blade-server system, the cabinet that holds all the server blade modules has two, three, four or more power supplies. The cabinet can sustain one or two power failures and still operate all of the server blades in the chassis using the surviving power modules. Fans to cool the servers and blade cabinet also need to be redundant; similarly, the cabinet itself houses most of the fans and has extra fans for redundancy purposes.

Network

Each server should have multiple network interface cards (NICs) installed or embedded on the motherboard. Multiple NICs are used for balancing traffic to achieve more performance as well as for redundancy should one NIC fail. You can use additional NICs to cre-

ate supplementary subnetworks to keep backup and recovery or management traffic off of the production network segments. In some systems, the NICs are actually installed within the shared server cabinet rather than on each individual server blade; this affords virtual mapping flexibility and redundancy, which are both highly recommended.

Storage

If you plan to use internal hard drives in your servers, ensure that you are using a Redundant Array of Independent Disks (RAID) controller to both *stripe* data across multiple drives (I'll explain what this is shortly), or mirror your drives for redundancy purposes. If you boot from SAN-based storage volumes (recommended), have multiple Host BUS Adapters (HBAs) or virtual HBA channels (in a shared-server chassis) so that you have redundant connections to the SAN with greater performance. Be wary of using any local server hard drives, for data or for boot volumes, because they do not provide the performance, virtual mapping, redundancy, and scalability of shared or SAN-based disk systems. Again, as stated earlier, there are alternative storage systems that use large numbers of replicated, inexpensive disk systems, without a true RAID controller, to achieve similar redundancy capabilities but the cost-benefit, features, and a full comparison of storage systems is beyond the scope of this book.

Scalability and replacement

In a cloud environment, as the service provider you must continually add additional servers to provide more capacity, and also replace servers for repair or maintenance purposes. The key to doing this without interrupting your online running services is to never install an application onto a single physical server (and preferably not onto local hard drives). If that server were to fail or require replacing, the application or data would be lost, leaving you responsible for building another server, restoring data from backup, and likely providing your customers with a credit for the inconvenience.

Key Take-Away

Using a SAN for OS boot volumes, applications, and data is recommended. Not only is the SAN significantly faster than local hard drives, but SAN systems are built for massive scalability, survivability, backup and recovery, and data replication between datacenters. This also makes it possible for the new blade servers to automatically inherit all of its storage and network mappings. With no configuration of the new or replacement server needed, the blade automatically maps to the appropriate SAN and NICs and immediately boots up the hypervisor (which then manages new VMs or shifts current VMs to spread workloads).

Server virtualization

Installing a hypervisor onto each physical server provides for the best utilization of the hardware through multiple VMs. As the cloud provider, you need to determine which hypervisor software best meets your needs and cost model. Some hypervisors are more mature than others, having more APIs and extensibility to integrate with other systems such as the SAN or server hardware management systems. The key to virtualization, beyond squeezing more VMs into each physical server, is the ability to have VMs failover or quickly reboot on any other available physical server in the farm. Depending on the situation and hypervisor's capability, you can do this without a customer even noticing an outage. With this capability, you can move all online VMs from one server to any other servers in the farm, facilitating easy maintenance or repair. When you replace a failed server blade or add new servers for capacity, the hypervisor and cloud management system recognizes the additional physical server(s) and begins launching VMs on it. (Figure 1-5 shows the physical servers that would run hypervisors and host guest or customer VMs.)

Figure 1-5 shows a notional example of a private cloud installed into a single physical equipment rack. The configuration includes:

- Two network switches for fiber Ethernet and fiber SAN connection to the datacenter infrastructure
- Three cloud management servers that will run the cloud management software platform
- A SAN storage system with seven disk trays connected through SAN switches to server chassis backplane
- Two high-density server chassis, each with 16-blade servers installed, running your choice of hypervisor and available as customer VMs (also called capacity VMs)

Additional expansion cabinets would be installed next to this notional cloud configuration with extra capacity servers and storage. Cloud management servers do not need to be repeated for every rack, but there is a limit, depending on the cloud management software vendor you choose and the number of guest/capacity VMs. When this limit is reached, additional cloud management servers will be needed but can be federated—meaning that they will be added under the command and control of the central cloud management platform and function as one large cloud that spans all of the expansion racks.

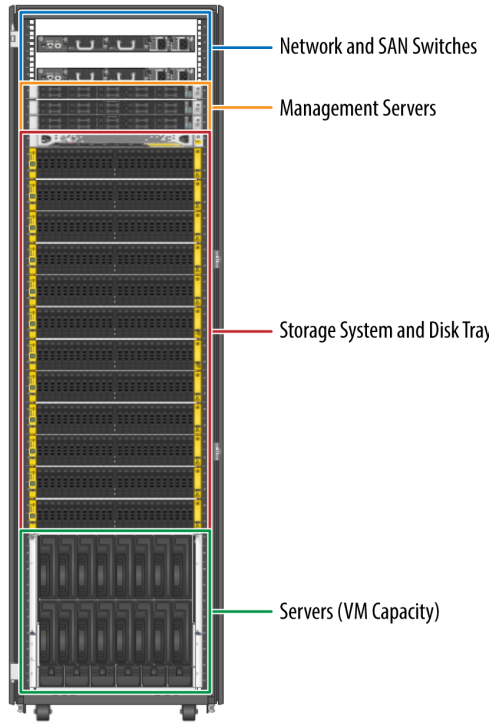

Figure 1-5. A notional private cloud in a single equipment cabinet

STORAGE SYSTEMS

Storage for large datacenters and cloud providers has come a long way from the days of simple hard drives installed within each server. This is fine for desktop workstations, but the performance of any one hard drive is too slow to handle hundreds or thousands of users. Modern disk drives are faster and certainly hold more data per physical hard drive than ever before, but do not confuse these newer hard drives with a true datacenter storage system. Even solid-state drives (SSD) are not always as fast as the multiple, striped disk drives that a SAN provides. Of course, combining striping and SSD provides the best disk performance—but at significant cost.

Anatomy of a SAN

The SAN consists of one or more head units (centralized "brains") that manage numerous trays of disk drives (refer back to Figure 1-5). A single SAN can hold thousands of physical disk drives, with the head units managing all of the strip-

ing, parity, cache, and performance management. As you need more storage capacity, you simply add more trays to the system, each one holding 8 to 20 drives. Most large SANs can scale from less than one, to six or eight racks full of disk trays. When one SAN head unit (or pair of head units for redundancy) reaches its recommended maximum number of disk drives or performance threshold, you can add additional head units and drive trays, forming another SAN in its own right. The management of multiple SANs from the same manufacturer is relatively easy because they use the same software management tools and can even make multiple SANs appear as one large SAN.

Within a SAN, there are often multiple types of disk drives. The cheapest ones used are SATA (serial AT attachment) drives. Although these types are the slowest, the trade-off is that they usually have the highest raw capacity. The next level up in performance and price is SAS (serial attached SCSI) drives; however, SAS drives do not have as much capacity as SATA drives. The next higher level of performance is from Fiber Channel disk drives but these are quickly being phased out due to cost and size limitations—SAS being a better mid-tier disk option in many cases. The premium level disk drives for performance and cost are SSDs. Depending on your performance and capacity needs, you can configure a SAN with one or all of these drive types. The SAN head units can automatically spread data across the various disk technologies to maintain optimum performance, or you can manually carve up the disks, striping, and RAID levels to meet your needs.

The latest SAN systems have an additional type of temporary storage called a cache. Cache is actually computer memory (which is even faster than SSDs) that temporarily holds data until it can be written to the physical disks. This technology can significantly improve the SAN performance, especially when pushed to its maximum performance limits. Some SAN manufacturers are now beginning to offer pure memory-based storage devices for which there are no actual disk drives. These are extremely fast but also extremely expensive; you need to consider if your servers or applications can actually benefit from that much performance.

Here are some factors you should consider when selecting and deploying storage infrastructure:

SAN sizing and performance

When selecting a SAN model and sizing, consider the servers and applications that your customers will use. Often the configuration of the disks, disk groups, striping, RAID, size of disk drives, and cache are determined

based on the anticipated workload the SAN will be servicing. Each individual SAN model will have a maximum capacity, so multiple sets of head units and drive trays might be needed to provide sufficient capacity and performance.

I highly recommend utilizing the SAN manufacturer's expertise to help you pick the proper configuration. Inform the SAN provider of the amount of usable storage you need and the servers and applications that will be using it; the SAN experts will then create a configuration that meets your needs. All too often, organizations purchase a SAN and attempt to configure it themselves, ending in poor performance, poorly configured RAID and disk groups, and wasted capacity.

Key Take-Away

Most cloud providers and internal IT organizations tend to overestimate the amount of initial storage required and the rate at which new customers will be added. The result is over-purchase of storage capacity and increased initial capital expenditure. Careful planning is needed to create a realistic business model for initial investment and customer adoption, growth, and migration to your cloud.

Fibre Channel network

There are various types of cabling, networks, and interfaces between the SANs and servers. The most popular is a Fibre Channel network, consisting of Fibre Channel cables connecting servers to a Fibre Channel switch. Additional Fibre Channel cables are run from the switch to the SAN. Normally, you have multiple fiber connections between each server with a switch for increased performance and redundancy. There are also between two and eight Fibre Channel cables running from the switch to each SAN, again providing performance, load balancing, and redundancy. You can also connect Fibre Channel network switches to additional fiber switches to create a distributed SAN environment, connect to Fibre Channel backup systems, and even implement replication to another SAN.

There are additional cabling and network technologies to connect servers to a SAN. iSCSI and FCoE are two such technologies, utilizing traditional network cabling and switches to transmit data between servers and SANs. To keep disk traffic separate from production users and network applications,

additional NICs are often installed in each server and dedicated to the disk traffic.

RAID and striping

There are numerous RAID techniques and configurations available within a SAN. The optimum configuration is best determined by the SAN manufacturer's experts, because every model and type of SAN is unique. Only the manufacturers really know their optimal combination of disk striping, RAID, disk groups, and drive types to provide the required capacity and performance—the best configuration from one SAN manufacturer will not be the same for another SAN product or manufacturer. Here are some key recommendations:

Striping

Striping data across multiple disk drives greatly speeds up the performance of the disk system. To provide an example, when a chunk of data is saved to disk, the SAN can split the data onto 10 striped drives as opposed to a single drive held within a physical server. The SAN head unit will simultaneously write one tenth of the data to each of the 10 drives. Because this occurs at the same time, the chunk of data is written in one-tenth the amount of time it would take to write the data to a single nonstriped drive.

RAID

I won't cover all the definitions and benefits of each RAID technique; however, I should note that SAN manufacturers have optimized their systems for certain RAID levels. Some have even modified a traditional RAID level, and essentially made a hybrid RAID of their own to improve redundancy and performance. One common mistake untrained engineers often make is to declare that they need RAID 10 —a combination of data striping and mirroring—for the SAN in order to meet performance requirements. The combination of striping and mirroring defined in a RAID 10 configuration does give some performance advantages, but it requires twice the number of physical disks. Given the complexity of today's modern SANs, making the blind assumption to use RAID 10 can be a costly mistake. Allowing the SAN to do its job, with all its advanced striping and cache technology, will provide the best performance without the wasted drive space that RAID 10 requires. Essentially, RAID 10 was nec-

essary years ago, when you used less expensive, lower-performing disks and had to maximum performance and redundancy; SAN technologies are now far better options to provide even more performance and redundancy along with scalability, manageability, and countless other features.

Key Take-Away

When using a modern SAN, making the blind assumption to use RAID 10 can prove to be a costly mistake. Allowing the SAN to do its job, with all its advanced striping and cache technology, will provide the best performance without the wasted drive space. Each SAN device will have its own recommended RAID, striping, and caching guidelines for maximum performance, so traditional RAID concepts might not provide the best results.

Thin provisioning

Thin provisioning is now a standard feature on most modern SANs. This technology essentially tricks each server into seeing *X* amount of disk capacity without actually allocating the entire amount of storage. For example, a server might be configured to have a 100 GB volume allocated from the SAN, but only 25 GB of data actually stored on the volume. The SAN will continue to inform the server that it has a total of 100 GB of available capacity, but in actuality, only 25 GB of data is being used; the SAN system can actually allow another server to utilize the free 75 GB. When done across an entire server farm, you save a huge amount of storage space by providing the servers with only the storage they are actually using. One important factor is that cloud providers must monitor all disk utilization carefully so that they don't run out of disk capacity because they have effectively "oversubscribed" their storage allocations. When actual utilized storage begins to fill up the available disk space, they must add disk capacity.

De-duplication

When you consider the numerous servers, applications, and data that utilize a SAN, there is a significant amount of data that is duplicated. One obvious example is the OS files for each server or VM; these files exist on each and every boot volume for every server and VM. De-duplication technology within the SAN keeps only one copy of each data block, yet essentially tricks each server into thinking it still has its own dedicated storage volume. De-duplication can easily reduce your storage requirements by a factor of 5 to 30 times, and that is before using any compression technol-

ogy. Critics of de-duplication claim it slows overall SAN performance. Manufactures of SANs offering de-duplication are, of course, aware of this criticism, and each have put in technologies that mitigate the performance penalty. Some SAN manufacturers claim their de-duplication technology, combined with caching and high-performance head/logic units are sophisticated enough that there are zero or unnoticeably small performance penalties from enabling the technology, thus making the cost savings appear even more attractive.

Key Take-Away

Consider using thin provisioning and data de-duplication when possible to reduce the amount of storage required. By embedding thin provisioning and de-duplication functionality within the chipsets of the SAN head units, most SANs now suffer little or no performance penalty when using it. Any penalty you might see is more than acceptable, given the amount of disk space you can potentially save.

Reclamation

When a VM boots up its OS, it allocates a certain amount of temporary swap or working disk storage capacity (e.g., 20 GB) to operate. When the VM is no longer running, it no longer needs this temporary working space from the SAN. The problem is that the data for the inactive VM still exists on the SAN. To reclaim that space, a process known as *reclamation* is used. Most SANs can perform this function, some automatically, and some requiring a software application to be manually executed or using a scheduled batch job. This technology is crucial, or you will run out of available disk space because of leftover "garbage" data clogging up available SAN storage even after VMs have been turned off.

Snapshots and backup

SANs have the unique ability to take a snapshot of any part of the storage. These snapshots are taken while the disks are online and actively in use, and are exact copies of the data—taking only seconds to perform with no impact on performance or service availability. The reason snapshots are so fast is because the SAN is technically not copying all of the data; instead, it's using pointers to mark a point in time. The benefits of this technology are many; the cloud provider can take a snapshot and then roll back to it anytime needed. If snapshots are taken throughout the day, the data volume can be instantly restored back to any pointer desired in the case of

data corruption or other problems. Taking a snapshot and then performing a backup against it is also an improvement in speed and consistency compared to trying to back up live data volumes.

Replication

Replication of SAN data to another SAN or backup storage system is very common within a datacenter. SANs have technology embedded within them to initially seed the data to the target storage system, and then send only the incremental (commonly referred to as *delta*) changes across the fiber or network channels. This technique allows for replication of SAN data across wide area network (WAN) connections, essentially creating a copy of all data offsite and fully synchronized at all times. This gives you the ability to quickly recover data and provides protection should the primary datacenter's SAN fail or become unavailable. Replication of SAN data from one datacenter to another is often combined with redundant server farms at secondary datacenters. This way, these servers can be brought online with all of the same data should the primary servers, SAN, or datacenter fail. This SAN replication is often the backbone of geo-redundant servers and cloud storage operations, facilitating the failover from the primary to a secondary datacenter when necessary.

BACKUP AND RECOVERY SYSTEMS

As a cloud provider, you are responsible for not only keeping your servers and applications online and available to customers, but you must also protect the data. It is not a matter of if, but when, a computer or storage system will fail. As I discuss throughout this chapter, you should be purchasing and deploying your servers, storage, and network systems with redundancy from the start. It is only when redundancy and failover to standby systems fail that you might need to resort to restoring data from backup systems. If you have sufficient ongoing replication of your data (much preferred over depending on a restore from backup), the only occasion in which you need to restore from backups is when data becomes corrupt or accidentally deleted. Having points in time—usually daily at a minimum—to which you have backed up your data gives you the ability to quickly restore it.

Backup systems vary greatly, but traditionally consist of software and a tape backup system (see Figure 1-6). The backup software both schedules and executes the backup of data from each server and disk system and sends that data across the network to a tape backup system. This tape system is normally a large

multitape drive library that holds dozens or hundreds of tapes. All data and tapes are indexed into a database so that the system knows exactly which tape to load when a particular restore request is initiated.

Backup software is still a necessary part of the backup and restore process. It is normally a two-tier system in which there is at least one master or controlling backup software computer (many of these in large datacenters) and backup software agents installed onto each server. When the backup server initiates a timed backup (e.g., every evening during nonpeak times), the backup agent on each server activates and begins sending data to the target backup system. Backups can take minutes or hours depending on how much data needs to be transmitted. Often, they are scheduled so that a full backup is done once per week (usually on nonpeak weekends), and each day an incremental backup of only the changed data is performed. The problem with this is that you might have to restore from multiple backup jobs to get to the data you are trying to restore. Full backups are sometimes so time consuming that they cannot be performed daily. One solution is to use SAN replication and snapshot technology rather than traditional incremental/full backup software techniques.

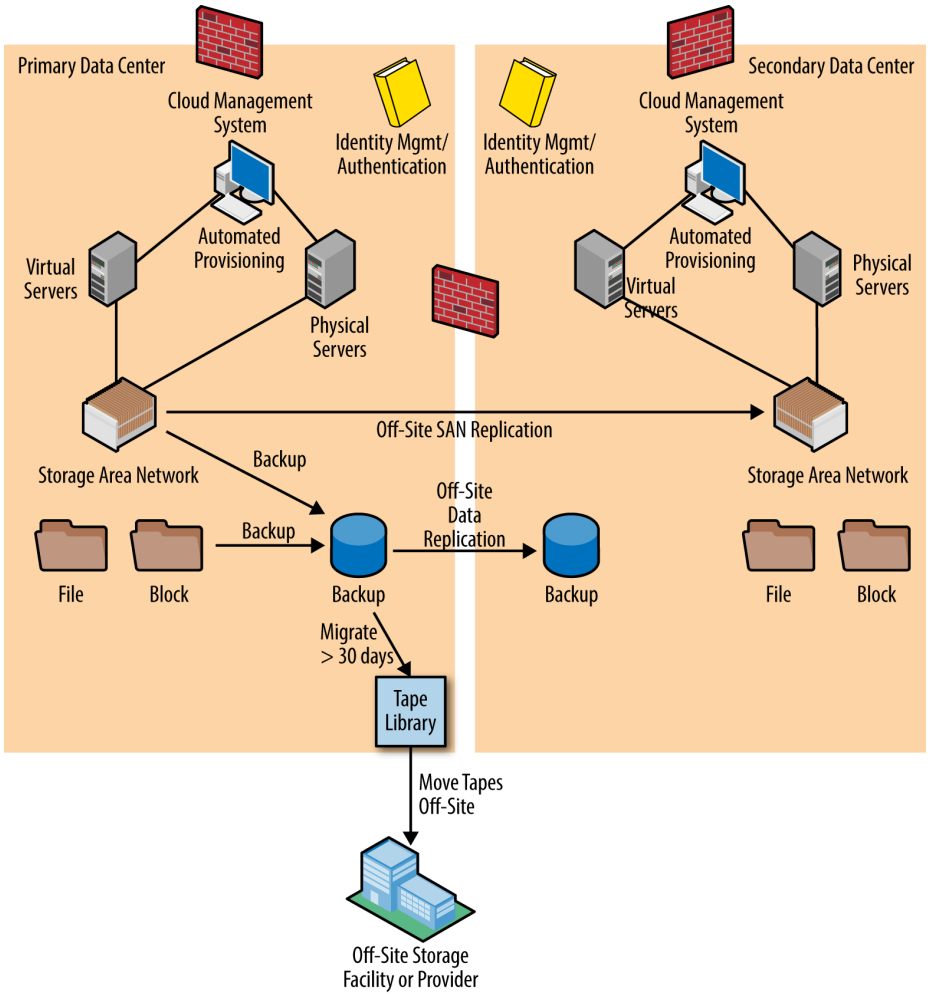

Figure 1-6. A traditional tape-based backup/recovery architecture

Modern backup systems are moving away from tape-based backup in favor of disk-based systems. Disk drives—particularly SATA-type drives—have become so inexpensive and hold so much data that in most cases they have a lower overall cost than tape systems. Disk drives provide faster backup and restoration than a tape system, and a disk-based system does not have the problem of degradation of the tape media itself over time; tapes often last only five to seven years before beginning to deteriorate, even in ideal environmental conditions. The next time

you have a customer demanding 10 or more years of data retention, advise them that older tapes will be pretty much worthless.

Key Take-Away

Modern backup systems are moving away from using tape-based backups in favor of de-duplicating, thin-provisioned, compressed disk-based backup systems. Many modern SANs now include direct integration with these disk-based backup systems.

Figure 1-7 shows SAN and/or server-based data initially stored in a backup system at the primary datacenter. This backup system can be tape, but as just explained, it's better to use a disk-based backup media. The backup system, or even a dedicated SAN system used for backup, then replicates data in scheduled batch jobs or continuously to the secondary datacenter(s). This provides immediate offsite backup safety and facilitates disaster recovery with standby servers in the secondary datacenter(s) when a failover at the primary datacenter occurs. There is often little value in having long-term retention at both datacenters, so a best practice is to hold only 14 to 30 days of backup data at the primary datacenter (for immediate restores), with the bulk of long-term data retained at secondary datacenter(s). This long-term retention could use tape media, but as explained earlier, this is not necessarily cheaper than disk storage, especially when you consider that tape degrades over time.

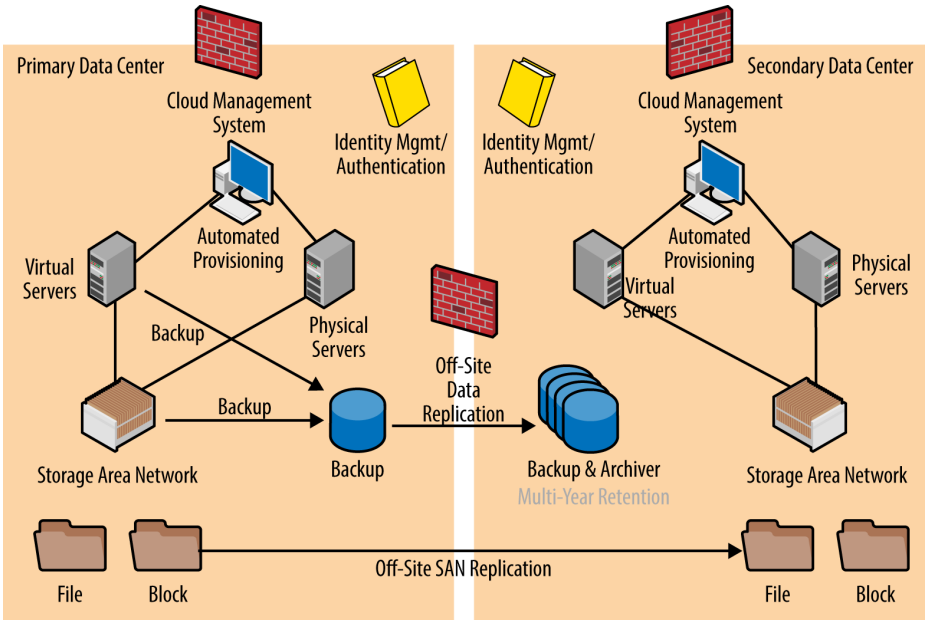

Figure 1-7. A modern disk-based backup/recovery architecture

Backing up VMs

A technique that is unique to virtualized server and SAN environments is the ability to back up VMs *en masse*. Rather than installing a backup agent onto every VM, the agent is only installed once on the physical server's hypervisor system. The backup agent takes an instant snapshot of each VM, and then the backup of the VM occurs based on this copy. This makes backups and restorations much faster.

Key Take-Away

New backup techniques and software that knows how to back up VMs in bulk (rather than per-VM software backup agents) is critical to success in backing up the cloud environment.

Replication of data, often using SAN technologies, is preferred and is now the new "gold standard" for modern datacenters compared to traditional backup and recovery. Given SAN capabilities such as replication, de-duplication, thin provisioning, and snapshots, using traditional backup tapes or backup software agents is no longer economical or desirable.

SOFTWARE SYSTEMS

Cloud providers need dozens of software systems to operate, monitor, and manage a datacenter. Servers typically have a hypervisor system installed with an OS per VM, with applications installed within each one. These software systems are what most consumers of the cloud are aware of and use daily. Other software systems that need to be deployed or considered include the following:

Security

Security software systems range from network firewalls and intrusion detection systems to antivirus software installed on every server and desktop OS within the datacenter and customer end-computing devices. It is also essential to deploy security software that gathers event logs from across the datacenter looking for intrusion attempts and unauthorized access. There are also physical security systems in place, such as cameras and biometric identity systems that you must manage and monitor. Aggregation and correlation of security events from across the system is now more critical than ever before when consolidating applications and data into a cloud environment.

Network

Network management software is used to both manage routers, switches, and SAN or converged fabric devices, as well as to monitor traffic across the networks. This software provides performance trending and alerts to any device failures. Some network software is sophisticated enough to automatically scan the network, find all network devices, and produce a computerized map of the network. This is very useful not only for finding every device on the network, but also when troubleshooting a single device that has failed and might be affecting an entire section of the infrastructure. As in security, the correlation of multiple events is critical for successfully monitoring and managing the networks.

Backup and recovery

Backup software is essential in any large datacenter to provide a safe copy of all servers, applications, and data. Hierarchical storage systems, SAN technologies, and long-term media retention are all critical factors. Integrating this into the overall datacenter management software tools provides better system event tracking, capacity management, and faster data recovery. As described earlier, consider backup software that integrates with

SAN and VM technologies—legacy backup software is often not suitable for the cloud environment.

Datacenter systems

A large modern datacenter has numerous power systems, fire suppression systems, heating and cooling systems, generators, and lighting controls. To manage and monitor everything, you can deploy software systems that collect statistics and statuses for all of the infrastructure's machinery. The most advanced of these systems also manage power consumption to allow for long-term capacity planning as well as to identify power draws. Although rare at this time, future datacenters will not only monitor power and environmental systems, but also utilize automated floor or ceiling vents to change airflow when a rack of servers is detected as running hotter than a set threshold. This type of dynamic adjustment is vastly more cost effective than just cranking up the cooling system. Some of the newer "green" datacenters utilize a combination of renewable and power-grid energy sources, dynamically switching between them when necessary for efficiency and cost savings.

CLOUD MANAGEMENT SYSTEM

The key purposes of a cloud management system are to provide the customer a portal (usually web-based) to order cloud services, track billing, and automatically provision services that they order. Sophisticated cloud management systems will not only provision services based on customer orders, but also can automatically update network and datacenter monitoring and management systems whenever a new VM or software application is created.

Key Take-Away

A cloud provider cannot operate efficiently without a cloud management system. Without the level of automation that a management system provides, the cloud provider would be forced to have so much support staff that it could not offer its services at a competitive price.

Cloud management systems are so important a topic, with significant issues and flexible options, that ??? has been dedicated to this subject.

REDUNDANCY, AVAILABILITY, CONTINUITY, AND DISASTER RECOVERY

Modern datacenters—and particularly cloud services—require careful consideration and flexible options for redundancy, high availability, continuity of opera-

tions, and disaster recovery. A mature cloud provider will have all of these systems in place to ensure its systems stay online and can sustain simultaneous failures and disasters, without customers even noticing. Cloud service quality is measured through service-level agreements (SLAs) with your customer, so system outages, even if small, harm both your reputation as well as your financials. People often confuse the terms "redundancy," "high availability," "continuity," and "disaster recovery," so I have defined and compared them in the following list:

Redundancy

Redundancy is achieved through a combination of hardware and/or software with the goal of ensuring continuous operation even after a failure. Should the primary component fail for any reason, the secondary systems are already online and take over seamlessly. Examples of redundancy are multiple power and cooling modules within a server, a RAID-enabled disk system, or a secondary network switch running in standby mode to take over if the primary network switch fails.

For cloud service providers, redundancy is the first line of protection from system outages. As your cloud service grows in customer count and revenue, the value of network, server, and storage redundancy will be obvious when you experience a component failure.

High availability

High availability (HA) is the concept of maximizing system uptime to achieve as close to 100% availability as possible. HA is often measured by how much time the system is online versus unscheduled outages—usually shown as a percentage of uptime over a period of time. Goals for cloud providers and customers consuming cloud services are often in the range of 99.95% uptime per year. The SLA will determine what the cloud provider is guaranteeing and what outages, such as routine maintenance, fall outside of the uptime calculation.

For purposes of this section, HA is also something you design and build into your cloud solution. If you offer your customer 99.99% uptime, you are now looking at four minutes of maximum outage per month. Many VMs, OSs, and applications will take longer than this just to boot up so HA configurations are necessary to achieve higher uptime requirements.

Key Take-Away

To keep your systems at the 99.99% level or better, you must design your system with redundancy and HA in mind. If you are targeting a lesser SLA, disaster recovery or standby systems might be adequate.

You can achieve the highest possible availability through various networking, application, and redundant server techniques, such as the following:

- Secondary systems (e.g., physical or VMs) running in parallel to the primary systems—these redundant servers are fully booted and running all applications—ready to assume the role of the primary server if it were to fail. The failover from primary to secondary is instantaneous, and causes no outages nor does it have an impact on the customer.
- Using network load balancers in front of servers or applications. The load balancer will send users or traffic to multiple servers to maximize performance by splitting the workload across all available servers. The servers that are fed by the load balancer might be a series of frontend web or application servers. Of equal importance is that the load balancer skip, or not send traffic to a downstream server, if it detects that the server is offline for any reason; customers are automatically directed to one of the other available servers. More advanced load-balancing systems can even sense slow performance of their downstream servers and rebalance traffic to other servers to maintain a performance SLA (not just an availability SLA).
- Deploy clustered servers that both share storage and applications, but can take over for one another if one fails. These servers are aware of each other's status, often sending a heartbeat or "are you OK?" traffic to each other to ensure everything is online.
- Applications specifically designed for the cloud normally have resiliency built in. This means that the applications are deployed using multiple replicas or instances across multiple servers or VMs; therefore, the application continues to service end users even if one or more servers fail. Chapter 2 covers cloud-native applications in more detail.

Figure 1-8 illustrates an example of an HA scenario. In this example, a VM has failed and secondary VMs are running and ready to immediately take over operations. This configuration has two redundant servers—one in the same datacenter on a separate server blade and another in the secondary datacenter. Failing-over to a server within the same datacenter is ideal and the least likely to impact customers. The redundant servers in the secondary datacenter can take over primary operations should multiple servers or the entire primary datacenter experience an outage.

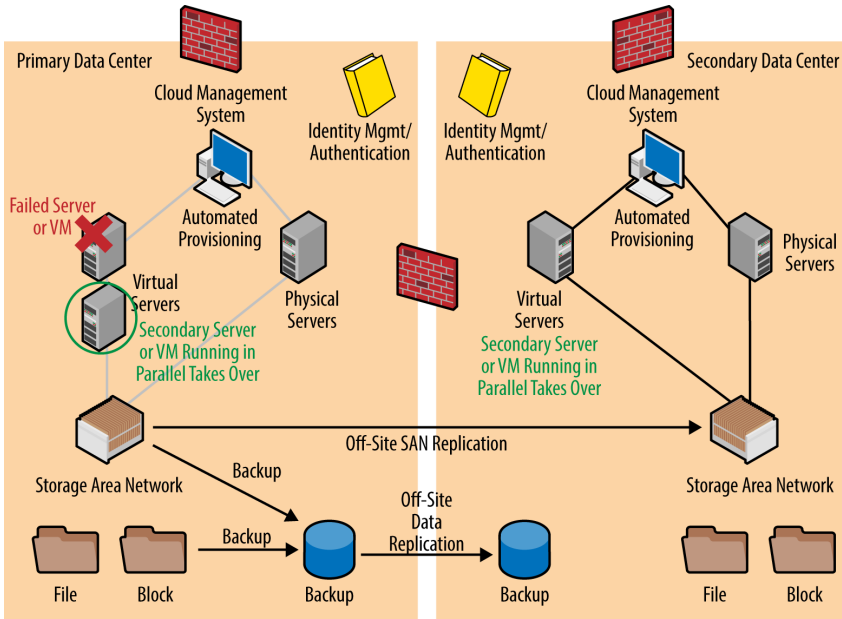

Figure 1-8. Example of HA: a failover

Continuity of operations

Continuity of operations (CoO) is the concept of offering services even after a significant failure or disaster. The dictionary definition is more generic, stating CoO is the ability to continue performing essential functions under a broad range of circumstances. For the purposes of being a cloud provider, CoO is a series of failover techniques to keep network, servers, storage, and applications running and available to your customers. In the real world, CoO refers to a broader range of keeping your entire service online after a significant failure or disaster. Many cloud providers will specifically iden-

tify events that are of a more significant nature, such as natural disasters; some spell out different SLAs or exceptions to SLAs when major, unavoidable events occur, versus normal system failures that are common within a datacenter.

A continuity plan for a cloud provider would typically involve failing-over to a secondary datacenter should the primary datacenter become unavailable or involved in a disaster. The network infrastructure, server farms, storage, and applications at the secondary datacenter are roughly the same as those in the primary, and most important, the data from the primary datacenter is always being replicated to the secondary. This combination of having prestaged infrastructure and synchronized data is what make it possible for you, as the cloud provider, to move all services to the secondary datacenter and resume servicing your customers. The failover time in such a scenario is sometimes measured in hours, with the best, most advanced environments failing-over within minutes. This CoO failover might not be as immediate as in a true HA configuration. If you can failover to a secondary datacenter and still function within your guaranteed SLA, that is, by definition, a successful CoO plan and execution.

Another part of a continuity plan deals with your staff and support personnel. If you must failover to a secondary datacenter, how will you adequately manage your cloud environment at that time? Will your staff be able to work from home if the primary office or datacenter location is compromised? The logistics and business plans are a huge part of a complete continuity of operations plan—it isn't only about the technology.

Disaster recovery

Similar to continuity of operations, disaster recovery (DR) is both a technology issue and a logistics challenge. As a cloud provider, you must be prepared for a disaster that could involve a portion or all of your datacenter and systems. When building your cloud system, you typically have two or more datacenters so that you can failover between your primary and secondary in the event of a disaster. If your cloud system design is more advanced, you might have three or more datacenters, all peers of one another, with none of them acting as the "prime" center. If an outage occurs at one, the others immediately take over the customer load without a hiccup (essentially, load balancing between datacenters).

A DR plan is similar to a CoO plan, but with one important addition. As a result of your CoO plan and technologies, you can continue to service your customers. A DR plan also includes how to rebuild the datacenter, server farms, storage, network, or any portion that was damaged by the disaster event. In the case of a total datacenter loss, the DR plan might contain strategies to build a new datacenter in another location, or lease space from an existing datacenter and purchase all new equipment. The recovery, in this worst-case scenario, might take several months to resume normal operations.

This leads to another aspect of your DR plan: while you are now running in your secondary datacenter, are the systems the same size and performance as your original, now-failed, primary datacenter? Maybe you have slower systems or storage with less capacity in your secondary datacenter; this is actually fairly common, because most providers assume that they will never remain operational within a secondary datacenter for very long before switching back to the primary.

Key Take-Away

The DR plan needs to document steps to bring the secondary datacenter up to its "primary" counterpart's standards in the event that there is no hope of returning operations back to the primary.

Figure 1-9 presents a scenario in which the primary datacenter has failed, lost connectivity, or is otherwise entirely unavailable to handle production customers. Due to the replication of data (through the SAN in this example) and redundant servers, all applications and operations are now shifted to the secondary datacenter. A proper CoO plan not only allows for the failover from the primary to secondary datacenter(s), but also documents a plan to either switch back all operations to the first datacenter after the problem has been resolved. Finally, as previously stated, the CoO plan should include procedures to make the secondary datacenter the new permanent primary datacenter should the first (failed) datacenter be unrecoverable.

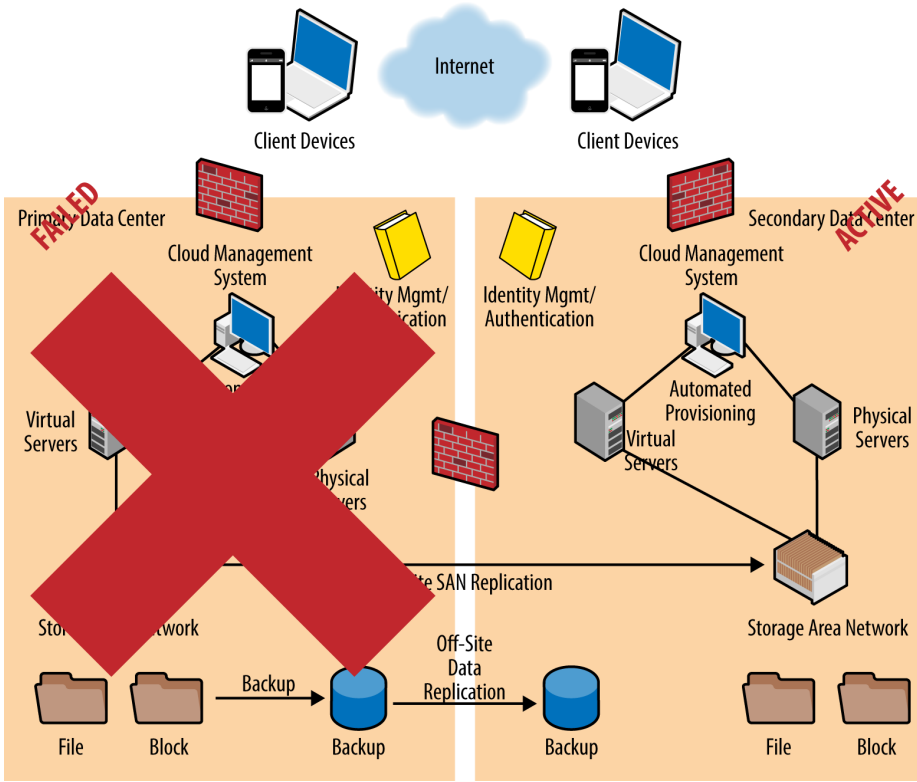

Figure 1-9. Example of disaster recovery—a continuity scenario

Managing Scope, Releases, and Customer Expectations

Lessons learned in the area of project scope management, schedules, and release management are not necessarily technical issues and not unique to cloud computing. This being said, experience teaches us that many transitions to a cloud environment failed or were significantly delayed due to mismanagement of customer expectations and scope creep.

SCOPE CREEP

Cloud technologies, portals, online ordering, and automated provisioning are new capabilities that customers begin to really appreciate and better understand when they begin to pilot and use the system. Past experience informs us that cloud customers very quickly begin expressing their desire for additional features and more customization—especially because on-demand, elastic, automatically

deployed cloud services is such a new capability. Just as in any IT project deployment, it is absolutely critical to manage customer expectations and stick with a solid proven plan for releasing a 1.0 version of the cloud service and then decide what customer requests can or should be accommodated through future releases of the cloud management platform portal or XaaS offering. Although any IT project has the potential for this scope creep, private cloud deployments seem to take this to an extreme because the cloud is a new style of IT service delivery for most customers. As the customer begins to realize the "art of the possible" the number and complexity of additional features and customization quickly gets out of hand—especially if the systems integrator or cloud provider has already bid and been awarded a contract with a set scope and price already established.

RELEASE MANAGEMENT AND CUSTOMER EXPECTATION MANAGEMENT

Controlling scope creep and customer expectations can be done through a release management plan and roadmap. Whether it is a public cloud provider with a published feature roadmap or a private cloud implantation, it is critical to control the scope and complete the initial 1.0 release of the environment. As the cloud is built or the transition of applications begins, it is a matter of how many (not if) changes and feature requests the customer(s) will make. Many of the requested changes are legitimate and should be added to a future release of the cloud service or management software—possibly added cost to the cloud service. Other customer requests must be carefully assessed and some politely rejected; for example, when the customer asks to implement a legacy technique that is not suitable for the automated cloud environment, or too specific a requirement that satisfies only one customer (particularly for public clouds).

One lesson among the many that experience has taught us repeatedly is that scope creep can be the death of a successful customized private cloud deployment. You can control customer expectations and scope by pushing new features to future releases, but before you officially commit to the future roadmap, evaluate the impact of these new capabilities on price and ROI. Other lessons learned are that cloud customers seem to forget that customization and new features costs money. The published or contract-stipulated pricing from the cloud provider or systems integrator is only accurate when the scope is managed and new feature requests are under control.

Deployment Best Practices

Based on lessons learned and experience from across the cloud industry, you should consider the following best practices for your organization planning.

CONSUME VERSUS BUILD

The basic acquisition question for any organization is whether to consume an existing service, usually from a public cloud provider, or to build and operate your own private cloud. Here are some considerations:

- You can usually consume services from an existing public cloud with little or no capital expenditures or term commitment. Private clouds must be configured and deployed and therefore often require capital expenditures or minimum commitments.
- Public clouds offer standardized services to a large quantity of customers and are therefore limited in customization. Private clouds can be significantly customized but often require an initial investment. Some cloud providers offer managed private cloud services in which the cloud is actually hosted and operated at the provider's facilities instead of the customer's datacenter.
- If you want to use both a private cloud and some public cloud services, you should strongly consider focusing on a private cloud management platform that has embedded hybrid or bursting capabilities to provision to public cloud providers—creating a hybrid cloud. With this configuration, you have a single cloud management platform in your private network that controls automated provisioning, aggregates billing, and manages operations for both your private cloud and any and all connected public clouds.
- Be mindful that if you purchase cloud services from multiple public providers, you can end up having different ordering and management portals for every provider. A hybrid cloud management platform is often better suited to handle both the internal private cloud all connected public cloud services with a unified customer portal.

CLOUD MODELS

Although most customers desire the pay-as-you-go elastic scalability and ease of management you get with public clouds, organizations with unique customization and security requirements must often consider deploying a private cloud (hosted within customer or third-party facilities). Here are some considerations:

- Lessons learned have shown that larger organizations and government customers benefit most from a private cloud. Soon after deployment, organizations then seek to use additional cloud services from other private, virtual-private, community, or third-party public cloud providers; thus, a hybrid cloud is quickly becoming the most prevalent model in the industry.
- Virtual private clouds (VPCs) have not seen the uptake that was originally expected in the industry. Customers either do not understand VPC or just don't want their private cloud and data hosted by a public cloud provider—essentially the definition of VPC. Thus, the use cases in which customers do want to have VPC seem to be limited. Public and private/hybrid clouds are still, by far, the primary forms of cloud being deployed. Where VPC really could shine is in the future community clouds in which shared capacity is available to multiple customers with the same level of security and requirements.
- Community clouds have not had a lot of adoption in the initial four to five years of the cloud industry. Peer organizations have had significant political and governance issues (e.g., policies, finances, procurement, operational control, individual fiefdoms) that have delayed or killed the community cloud initiative. The biggest concern between community cloud owners is that peer organizations have little or no control over one another—what happens if an organization loses funding or changes its mission in ways that jeopardize critical applications it once provided to the rest of the communities? Cloud brokering might turn out to be a better alternative for organizations wanting multiple cloud providers and aggregated cloud service portals.

DATACENTER INFRASTRUCTURE

Modernizing a legacy datacenter with virtualization or automation does not equal a cloud, but it is an excellent start. There are numerous best practices for modernizing a datacenter that also prepare the environment for cloud services. Here are some considerations:

- Many organizations have now realized that the expense of infrastructure, facility, and staff is too much to justify building or continuing to operate a datacenter. This is due in part to the fact that many datacenters have excess capacity and the perfect return-on-investment (ROI) model only works when a datacenter is completely occupied. Then, when you achieve full occupancy, you have little or no room to grow. Often, the best advice is to get out of the datacenter owner/operator business unless that is your core business offering to customers. If you are not an IT provider company, why take on such a large expense and responsibility?
- Organizations that do need a datacenter should consider leasing out a pod or section of capacity from an existing datacenter provider. You can lease bulk-rate power, cooling, and space at a fraction of the cost of a full datacenter and without the hassle of purchasing and operating the physical facility, equipment, and staff.
- Some organizations now find that pods or shipping containers that contain preinstalled servers and storage (a datacenter in a box) are a good value and can facilitate quick expansion.
- You should try to avoid using the legacy existing network infrastructure (routers, switches, fabrics, physical servers, tape backup media) for the new cloud for two primary reasons:
    - Legacy infrastructure systems are often not as capable of automation, software-defined configuration, multifabric/protocol, and higher densities and performance as modern cloud-enabled systems.
    - As you build and grow your new cloud, you want to avoid the need to seek approval from change control boards, legacy customers, and other operational IT departments—the legacy systems and processes often slow down the new cloud-based processes and configuration needs. After the cloud services are built in the datacenter, you

should consider the migration process of bringing legacy infrastructure and applications over to the cloud, not the reverse.

CLOUD INFRASTRUCTURE AND HARDWARE

When building your own cloud infrastructure, many components and technologies are similar to any modern datacenter. Although using the highest density servers and network infrastructure is clearly advantageous, there are technologies that do make some servers, storage, and networking hardware better able to support a cloud-enabled environment. Here are some considerations:

Server infrastructure

Focus on blade- or cartridge-based servers that install into a common chassis or cabinet—usually 10 to 20 servers fitting in each cabinet, with 3 to 4 cabinets per equipment rack. Look for providers that maintain multiple levels of redundancy in power, cooling, and modular components, such as network and SAN host adapters built into the shared cabinet. Software-based mapping of network, SAN fabric, and other shared cabinet features afford the capability to swap out server blades, which automatically inherit the virtual configuration of that slot; thus no reconfiguration of the network or storage mappings is required. Lights-out and out-of-band management and monitoring tools should also be embedded in the modular servers and shared cabinets. Unique features specific to the cloud also include the ability to perform bare-metal configuration and hot swapping of resources (storage, network adapters) that support a fully automated cloud environment.

SANs

Avoid local hard drives embedded on each server blade/cartridge—the benefits of booting from a shared storage device such as a SAN far outweigh local storage in terms of flexibility, performance, and reliability. Though some manufacturers might claim that having multiple cheap modular servers each with inexpensive local storage is a good thing, in reality each physical server or blade server now hosts multiple VMs and thus any failures have an impact on multiple applications and multiple customers. Ignore the manufacturer: local, cheap, and slow hard drives are not desired in an on-demand, elastic, automated cloud environment.

Sizing of infrastructure

Purchasing an overly large initial cloud infrastructure can result in such high capital expenses that the ROI model cannot show a profit for several years. So, start reasonably small with the infrastructure, but use modular and highly scalable server, storage, and networking equipment for which you can expand capacity by adding modules—automation will take care of configuration and installation of software in most cases. Negotiate quick terms and contracts with hardware vendors to have new equipment prestaged at no cost until utilized or at least shipped to you and ready to install within 24 to 48 hours from the point at which you make the call. Continuously monitor capacity on your infrastructure so that you can predict when new hardware is needed without over-purchasing excessive new capacity. Even if you can get vendors to supply prestaged, free-until-you-use-it servers and storage, don't forget that sometimes it costs money to power and cool these systems. However, with some of the newer hardware, you can take advantage of *autopower-on* capability with which you can keep the devices idle until they are needed—for truly on-demand capacity.

SANS

Not all SANs are the same. There are unique features that some SANs have—embedded virtual storage controllers and embedded hypervisor VM support—that specifically support a cloud environment. Here are some considerations:

- SANs provide the maximum amount of performance, reliability, scalability, and flexibility to connect disk volumes to your high-density server farms and applications. As stated earlier, avoid local disk drives embedded on each server: you need server blades to be interchangeable, so a software-defined mapping to the SAN storage is desired for both boot and data volumes.
- Utilize thin-provisioning features of your SAN, with which you can overallocate storage—a primary reason why shared virtualized storage costs less in a cloud environment. Note that VM hypervisors can also perform thin provisioning, and you can only have one or the other (the SAN) perform thin provisioning at the same time.
- Dismiss arguments that thin provisioning, de-duplication, snapshots, and other SAN-based features reduce overall performance; SAN controller units are far faster than disk I/O; therefore, these SAN features have

almost no impact on performance (or such a small impact that it's nearly impossible to measure), especially with flash-based caching turned on.

- Utilize a combination of flash cache, solid-state disks, and traditional disk drives within the SAN. SAN controllers are able to automatically move data, and when properly configured, provide you with the maximum performance possible. Follow the advice of the SAN manufacturer to get the best configuration of disk striping, RAID, flash cache, and LUN allocation —every SAN manufacturer's embedded software is unique, so be wary of SAN performance advice from legacy storage operations staff, especially when they are not experienced with the latest SAN hardware and configurations.
- Never assume that RAID 5—or any level of RAID configuration—is appropriate or will provide the best performance without checking with the SAN manufacturer. Countless lessons learned prove that every SAN manufacturer implements their own disk striping, caching, and internal algorithms to achieve maximum performance and redundancy. Assuming RAID 5, for example, would be similar to turning off four-wheel drive in an SUV because you think you can do a better job driving in snow than the intelligent all-wheel-drive system.
- Avoid splitting disks into too many volumes. There is a limit within every SAN on the number of volumes that you can configure. When using a VM hypervisor, one large volume can easily service dozens of VMs. A one-volume-to-one-VM design is an antiquated model and likely will limit performance.
- Do utilize SAN-based, point-in-time snapshots or backups for daily, or even hourly, checkpoints to which data can be restored (recovery points). Use snapshots and VM technology to perform backups of all VMs rather than individual backup software agents on each VM. Remember that snapshots do require some additional storage capacity but have little or no impact on performance when each snapshot is taken.
- Use SAN-based data replication—potentially in real time or based on snapshot intervals—to send a copy of data to another local or remotely located storage device for immediate offsite backup capabilities. This technology is also very useful when configuring multi-datacenter redundancy and con-

tinuity of operations, replicating data to a secondary datacenter(s) in near real time if desired.

CLOUD MANAGEMENT PLATFORM

All cloud providers, or operators of a private cloud, utilize a cloud management software system that provides a customer ordering portal, subscription management, automation provisioning, billing/resource utilization tracking, and management of the cloud. Here are some considerations:

- Always deploy the cloud management platform and as much of the automation as possible when initially building any cloud. Avoid the temptation to deploy the cloud management platform and automation at a later time —after the cloud infrastructure, hypervisor, storage, and VMs are installed. Experience teaches us that it is exponentially more difficult to deploy the cloud management platform and automation afterward. And although provisioning and all management is temporarily performed manually, all of the cloud ROI models for pay-as-you-go, capacity management, patching and updates, and support personnel labor are negated.
- You can perform integration with every backend network, server, security, service ticketing, or application management system in a secondary phase after initial cloud capabilities are brought online. This is often the reality, given many datacenters already have these operational management tools and staff that can slowly transition to the cloud. Lessons learned have shown that trying to integrate with every legacy datacenter management tool upon initial launch can significantly delay the go-live date for your new cloud.
- There should be an ongoing effort to improve the cloud portal, reports, and statistical dashboards. Begin with at least a basic cloud portal and consumer-facing dashboard but try to avoid scope creep when customers ask for so many new features that you end up missing your initial launch timing goals.

SCOPE AND RELEASE MANAGEMENT

Because the cloud is new to both your organization and the consumers of the cloud services, avoid blindly accepting every request for improvements and enhancements. Get the first version of your cloud launched and push new feature requests to future releases as part of a published roadmap. Here are some considerations:

- Avoid scope creep. As in an IT project, scope creep can and will get out of control if not managed. Every new feature request can have a significant impact on costs and the cloud infrastructure, so avoid making quick decisions and approval of new features to the roadmap until you have made a full and careful analysis.
- Customers will ask for feature after new feature in the cloud portal—a particular area of scope creep—so be careful and ensure that customers understand the cost of such enhancements, especially if you, as the cloud provider, have already bid on or published service pricing.
- Many of the new features and capabilities of your private cloud will be available from the manufacturer of the deployed cloud management platform. Thus, follow its roadmap and request new features from the cloud management platform manufacturer when possible rather than overly customizing the cloud platform, which could result in difficult future upgrades.
- Customers often have multiple "gold" or standardized operating system templates of images that they'd like to use with their cloud VMs. Always evaluate and test their images for suitability and remember, as a cloud provider, you now own all future updating, patching, and support of these new OSs and configuration. You might want to provide a certain number of image templates within your pricing plan, with additional cost to import, test, and support the lifecycle for every new OS or VM image your customer requests. Often, it is best to have a minimal number of OS templates and use the software distribution and automated installation tools to install variations, updates, and applications rather than trying to manage so many unique combinations of OS and application through templates.

USING EXISTING STAFF

In ???, in the Operational Lessons Learned section, I discuss the use and possible reorganization of existing operational personnel to support your new private cloud. In this section, I provide best practices for using existing operational staff for the deployment of your cloud. Here are some considerations:

- Existing operational personnel are often hired and expected to perform only routine tasks following runbooks based on the current legacy infrastructure and applications. These individuals are not usually the best personnel to employ for new cloud technologies, processes, and techniques (if they had all the required knowledge for cloud and automation, they likely wouldn't be in a lower-paid operations role).
- Existing operational personnel already have what should be a full-time job, so asking them to deploy and properly configure new systems using new techniques often results in poor build quality, poor configuration consistency, and old bad habits being brought into the new cloud environment.
- Engage new, highly skilled personnel—as employees or contractors—to perform the deployment and configuration of the new cloud hardware. Existing personnel can participate in the hardware staging of systems, as they might need to assume responsibility for operating these systems, but ensure that the primary configuration is performed by the new team with a handoff to the legacy operations staff further down the road.
- Use only new, highly skilled personnel on the cloud automation software platform configuration and ongoing continuous automation improvements. Legacy staffers usually do not have the skills or experience to configure these systems. Over time, you can train these people and have them work alongside the newer staff to automate all legacy manual processes.

SECURITY

Assessing, certifying, and managing a private cloud environment is not very unique compared to a modern datacenter. The new cloud portal might require further vetting and testing, especially if it accepts payments or can be accessed from the public Internet. The actual VMs, storage, and applications all operate from behind one or more firewalls. As a result, there are often few aspects that

are unique to the cloud for security personnel to certify at the infrastructure level. Here are some considerations:

- Security should focus on the configuration and how the automated cloud management system configures VMs and OSs, applies patches and updates, and manages the virtual networks to which each VM is configured to communicate. When the templates and network zones are preapproved, the automation software should maintain consistent and reliable configurations when ordered by customers.
- Security should evaluate and preapprove each "gold" or VM image along with its OSs and baseline patch levels. When approved, the automated cloud management platform will consistently deploy VMs based on these images.
- Continuous monitoring should already be a primary goal of the security operations team within the datacenter. The cloud brings in a new level of automation and consequently new VMs and applications brought online 24-7; therefore, all new systems should immediately be added to the change control logs and inventory systems, and trigger immediate scans and ongoing monitoring by security after the VMs are online.
- For additional best practices, read ???.

Application Transformation

Key topics in this chapter:

- Application evolution to the cloud
- Application categories and characteristics
- The new approach to application development and delivery
- Continuous application delivery automation
- Application transformation methodology
- Application operational considerations
- Application assessment
- Application transformation best practices

Although many clouds initially focused on Infrastructure as a Service (IaaS), enterprise organizations often have applications listed as their top business priority. The reality is that a private or public cloud requires a base infrastructure of server, storage, and networking in order to begin hosting anything—an IaaS, Platform as a Service (PaaS), or Software as a Service (SaaS). In earlier chapters, I covered architectures for baseline IaaS clouds, but the largest task in the journey from enterprise IT to the cloud is in application transformation.

Key Take-Away

IaaS virtual machine services are now considered the very minimum capability for a public or enterprise private cloud. Application transformation is the longer-term goal and will take the most amount of time.

Application transformation is a fancy term for assessing the current applications and then planning and conducting migrations. When planning for your cloud transition or deployment, assessing your existing applications will lead you to decisions on what to migrate to the cloud, what apps to redesign, which to maintain in the existing enterprise, and the priorities for an eventual transition to the cloud. Before planning any application transition, it is essential to understand the basic characteristics and features of traditional and cloud-native applications.

Evolving Your Applications for the Cloud

??? discusses the evolution of computers, including a graphic depiction in ???. Figure 2-1 repeats that illustration, but this time let's focus on the evolution of applications that are shown below the trend line.

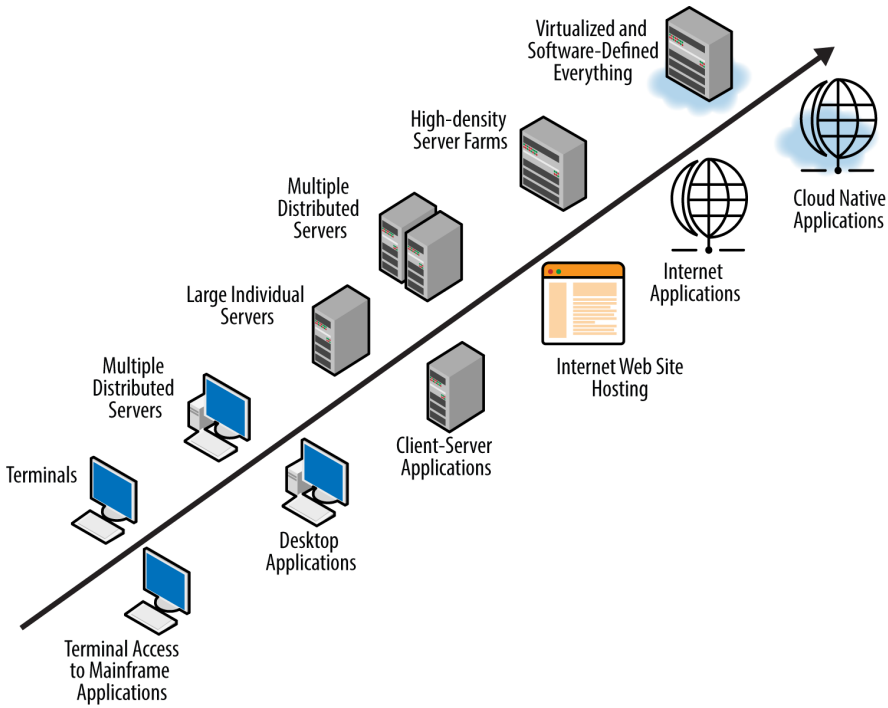

Figure 2-1. The evolution of cloud applications

Though computer hardware has evolved at a continuous and rapid pace, software traditionally has been slower to progress—until now. Although cloud computing is a new style of IT delivery, a new style of application architecture,

development, and delivery is upon us. The pace of application development and delivery of new features to consumers is now expected at a pace never seen before in the history of the IT industry. In fact, I will go as far as saying that software is now outpacing hardware in terms of innovation and impact to business and everyday life.

Coming back to Figure 2-1, there are some very important software architecture concepts of particular note:

- Desktop applications are still common but remain inefficient to support and upgrade. Remote desktops or Virtual Desktop Interface (VDI) is often still too expensive as well and still a compromise in ease of use for many organizations. Application publishing essentially pushes desktop applications into the cloud-enabled app category for at least a temporary bridge until a true cloud-enabled application is available.
- Client-server applications have been considered legacy since as early as 2010 (in case you didn't get your official notice in the mail). They are considered too chatty or noisy for efficient network communications and scalability, and they are expensive to deploy in high availability/resilient configurations. These applications are good candidates for *refactoring* (this term is defined later in this chapter) to the cloud.
- Internet applications are really a combination of multitiered client-server applications and web applications hosted on the Internet—this is the Internet service provider (ISP) and application service provider (ASP) era in the years up to 2009 before the cloud. These were rarely very efficient, but they were profitable for the providers and software developers and considered bleeding edge at the time. Most of the software languages, application programming interfaces (APIs), and tools used to create these applications are effectively dead. These applications, providers, and software companies either already upgraded to cloud applications and SaaS or went out of business because they didn't upgrade fast enough in the early days of the cloud.
- Cloud-enabled applications, also referred to as *cloud-native applications*, are designed to take advantage of a cloud infrastructure and all of its capabilities. Cloud-native is fully defined later in this chapter.

Application Categories

Applications can be categorized in many ways, but for this discussion, I will group them into four categories for clarity and relevancy in a cloud service environment:

COTS
: COTS applications are readily available from numerous software vendors, but their suitability for implementation in the cloud is not always straightforward. Many software vendors are rewriting their applications to both fit into a multitenant cloud environment and adapt their licensing models for pay-as-you-go cloud usage models; however, there are still software vendors that do not have properly designed cloud-native applications that work in a multitenant environment.

Open source
: These applications are often not shrink-wrapped or ready for purchase "on the shelf" such as with COTS products. As the name indicates, open source also means that the source programmed code is freely published so any party can customize it for its own needs, scan it for security vulnerability, and integrate other software components. Open source applications are often considered a community project with numerous (meaning hundreds or thousands) of developers, contributors, and integrators across the world. These open source applications are free for anyone to download and use, but they sometimes have licensing limitations such as not being able to use them for profit without permission or royalties paid. Some software companies take an open source application, enhance it with their own software, and sell their own distribution of the system. One key advantage is that open source applications often utilize industry standards and APIs or end up becoming a standard after sufficient adoption and industry acceptance. Besides the cost, one primary benefit to open source is that you avoid vendor lock-in. This is when you purchase and use a proprietary product and find that you are stuck with that product or it would be too costly to migrate away from it in the future.

Custom applications
: Also known as homegrown applications, these are what many large organizations have the most difficulty with when planning the transition to the cloud. These applications can range from legacy mainframe programs, all

the way to heavily modified COTS systems. Because there are often no integration standards and often little integration among these custom applications, there is also no single method of integrating or porting them to the cloud. Careful assessment and planning for each major application is required. A significant portion of this chapter is dedicated to this assessment topic.

Cloud native

Cloud native applications are designed to be hosted in a cloud environment and take advantage of a cloud infrastructure. This means the applications can, for example, remain online during planned or unplanned infrastructure outages and scale up or scale down to meet workload demands. Cloud applications are designed to work with other components and services within a cloud, such as databases, frontend web services, transaction queuing, payment engines, and so on, all working together in what appears to be a single cloud application or service. Cloud native apps consist of *composable* services (more on this term in the sidebar that follows) designed to be dynamic with respect to the infrastructure and other services with which they integrate—essentially discovering and dynamically registering services with other applications and components within the cloud rather than hardcoded mappings or static configuration. Cloud-native applications are developed to be elastic so that they can scale out automatically based on defined workload parameters. Finally, cloud applications are designed to be resilient so that the system can recover or self-heal when a problem is detected in the infrastructure, such as temporary hardware, software, or communications failures.

Composable Services and Applications

As the cloud continues to evolve, along with it has developed a small dictionary of new words and terms that, although they might not pop up in day-to-day conversation at the supermarket or while you're watching your kid's soccer game, they are common within the industry and often indispensable at getting an idea across concisely. *Composable* is one such term. Essentially, a composable service or application is one that is not hardcoded. Instead, it is flexible and adaptable, and can sense and detect its surroundings and nearby applications and services so that it could function in any cloud or environment in which it's functioning.

> Think of a composable as somewhat like creating a form using sculpting clay (whose shape is inherently flexible and responsive), as opposed to using wood or some other inflexible materials; when a wood sculpture is completed, it's nearly impossible to shape it into anything else, but a composable, like clay, can adapt to its environment and whatever new form that entails.

Application Characteristics

There are numerous characteristics that an application should have when transitioning to a cloud infrastructure. A public cloud provider will design these attributes into their application and cloud offerings, whereas a private cloud operator will have to assess and migrate legacy applications. Key characteristics of cloud-enabled (cloud-native) applications include the following:

Secure multitenancy
: This means that the application is configurable so that multiple customers can share the same instance of it, while maintaining separation of all data and user accounts. Customers and users within the shared application instance cannot see one another nor any data other than their own by using security access controls lists (ACLs), role-based permissions, and sometimes separation of databases (the application is still shared but can access and utilize separate databases when appropriate).

Elasticity
: Applications should be elastic as traffic, demand, and usage increases. Elastic means that an application can scale out (adding more compute resources or additional virtual machines) to handle an increase in workload or utilization. A properly designed cloud-native application should be able to automatically detect high utilization and trigger the necessary steps to start up new VMs or application services to handle peak workloads. Then, when peak utilization diminishes, it should reduce VMs or compute instances. Legacy applications that were not specifically designed to run in the cloud can often use a VM hypervisor to monitor processor and memory utilization, triggering more VM instances when a manually defined peak-utilization threshold is attained. These elasticity techniques make it possible for the applications to take advantage of the power of the cloud infrastructure to dynamically scale—even if the application wasn't origi-

nally designed as cloud-native; although, a cloud native application is more efficient.

Resiliency

Resiliency refers to the application's ability to survive and remain online during an infrastructure outage or failure. A properly designed cloud-native application would have multiple techniques to retry failed transactions and there would be multiple instances of the application services running on other servers or VMs. Legacy applications that were not designed for a cloud can use tools within a hypervisor or cloud infrastructure such as traffic load balancing, clustering, and server/VM failover, but these are not as effective and transparent to the end user as a cloud-native application's resiliency. There are many other aspects of resiliency and cloud-native application design benefits that will be covered later in this chapter.

Authentication systems

Applications that might have run within existing enterprise datacenters often utilized the internal corporate Microsoft Active Directory or some other identity management system to authenticate user logons. Ideally, applications hosted in a cloud should not assume Active Directory or the internal identity system is available; instead, they should favor an industry standard for authentication and directories such as Lightweight Directory Access Protocol (LDAP) or Security Assertion Markup Language (SAML). Both provide authentication capabilities—SAML is a bit more robust and appropriate as part of a single sign-on (SSO) system.

Universal access

The applications must be accessible from anywhere on the Internet or other wide area network (WAN) circuit. The application should be compatible with any and all access methods, from web-based access, virtual private network (VPN) access, and every variety of desktop, notebook, tablet, and mobile device. This also means that the user data must also be immediately available when moving from one computing device to another. As a general rule, you should follow the same philosophy as software-defined networking: never hardcode any network addresses or assume anything; always assume applications, users, and data could exist or operate from anywhere in the world through any network connection and any form of user interface or device (see also the section "Mobility" that follows momentarily).

Multitiered applications and platforms

Many cloud applications employ a multitiered design wherein there are multiple layers, or tiers, of services. These tiers separate the backend database, middleware and applications, and frontend processing. This tiered design facilitates higher levels of security, application upgrade modularity, and independent scalability of individual tiers as needed, based on utilization. Tiered applications also benefit from shared platform or PaaS applications in the cloud, such as a database service, that multiple applications can share for lower cost of maintenance, licensing, and operations.

Mobility

With the increasing number of applications hosted in a cloud environment, users or consumers often use mobile computing devices such as tablets or smartphones. The legacy assumption that end users only have desktop PCs or a particular PC operating system (OS) is no longer true. The concept of *mobile first* was adopted over the past few years but is more recently replaced with *ubiquitous access*—the intention of both being that applications need to be designed with the ability for users to access the system through any form of computer device, from any location, and have the same experience. Mobility might also require additional security and asset configuration management features and tools to ensure identity, data encryption, data privacy, and synchronization to mobile devices for offline viewing.

Key Take-Away

Ubiquitous access, elasticity, resiliency, and persistent data are the keys to successful cloud-native applications. Applications and data must always be accessible, from any form of computing device and any location, and with a consistent user experience.

As described later in this chapter, you should consider these characteristics when developing any new applications that you plan to host in the cloud. Also, evaluate your current applications to determine if any of the features are already embedded or how legacy applications could be modified to incorporate them.

The New Approach to Application Development and Delivery

Today's newest and most modern approach to application development is focused on rapid and continuous delivery. Private organizations and software

developers are now considered "legacy" and uncompetitive, following previously traditional practices such as releasing applications once or twice a year. It is not just a trend but an expectation of business users to get more frequent updates and new software releases—as often as quarterly or monthly. Many commercial software manufacturers are already using software release cycles measured in weeks if not days, launching not just bug fixes but true new features regularly rather than saving up these enhancements for a big "annual product launch."

Now, enter the public cloud providers and their software applications. Cloud providers have used their infrastructure and automation to further increase the pace of software delivery, sometimes daily if not multiple times per day, particularly because there are so many cloud offerings and subcomponents within each offering. The point here is that the automation and cloud management systems used by the public cloud providers can also be used in a private or enterprise cloud to facilitate rapid application development, testing, and continuous pushing to production as quickly as you and your organization can muster. Figure 2-2 compares notional application delivery cycles for a traditional application to a cloud-based continuous delivery application (future state).

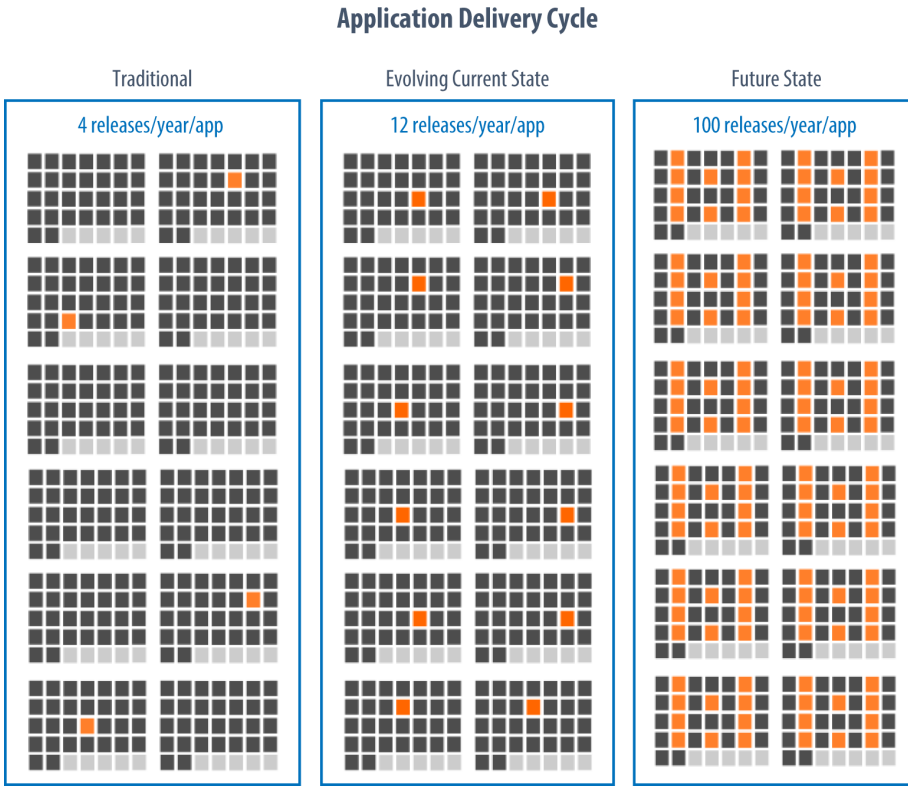

Figure 2-2. Notional application delivery cycle—traditional and with continuous delivery

Welcome to the new style of IT—developer style!

Figure 2-3 presents a comparison of traditional application development compared to cloud-based development. Technically, this new pace and style of application development is not specific to the cloud; however, the cloud does provide unique capabilities such as elasticity, software-defined networking, on-demand Development/Testing as a Service (Dev/Test), shared application lifecycle platforms and tools, and an enormous worldwide hybrid infrastructure. Many of the other cloud-based development characteristics shown in the figure were described earlier in this chapter.

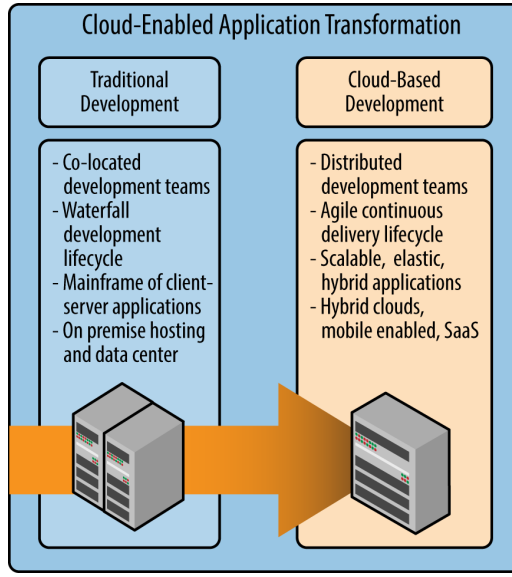

Figure 2-3. Cloud-based application development

CONTINUOUS APPLICATION DEVELOPMENT AND DELIVERY

Although not unique to the cloud, the term *continuous delivery*, sometimes called *continuous application development*, is now considered the modern methodology for application development and delivery. Continuous delivery employs an Agile-type approach to deliver continuous small software releases into production. To facilitate this application development lifecycle, continuous delivery in a cloud environment takes advantage of automation and software-defined networking to speed application development and promotion to production.

Through the use of VMs and cloud automation, new development environments—consisting of one or more VMs—you order, can provision services on-demand within minutes. You can create multiple sets of VMs so that multiple development teams can work in parallel or on different versions of the same application. Using VM snapshots and automation, these development VMs can be copied to a separate network segment/subnet for testing purposes while the original VMs remain intact for the developers to continue their work. Again, using snapshot technology, you can run multiple testing scenarios and then reset the application back to the previous snapshot for more testing. Finally, you can promote the test VMs or copy them to a production network or subnet through the application development automation—thereby launching the new application

for production users. This lifecycle is repeated over and over again while also allowing continuous development and testing activities on the next application release. Figure 2-4 offers an overview of the continous application development and delivery process.

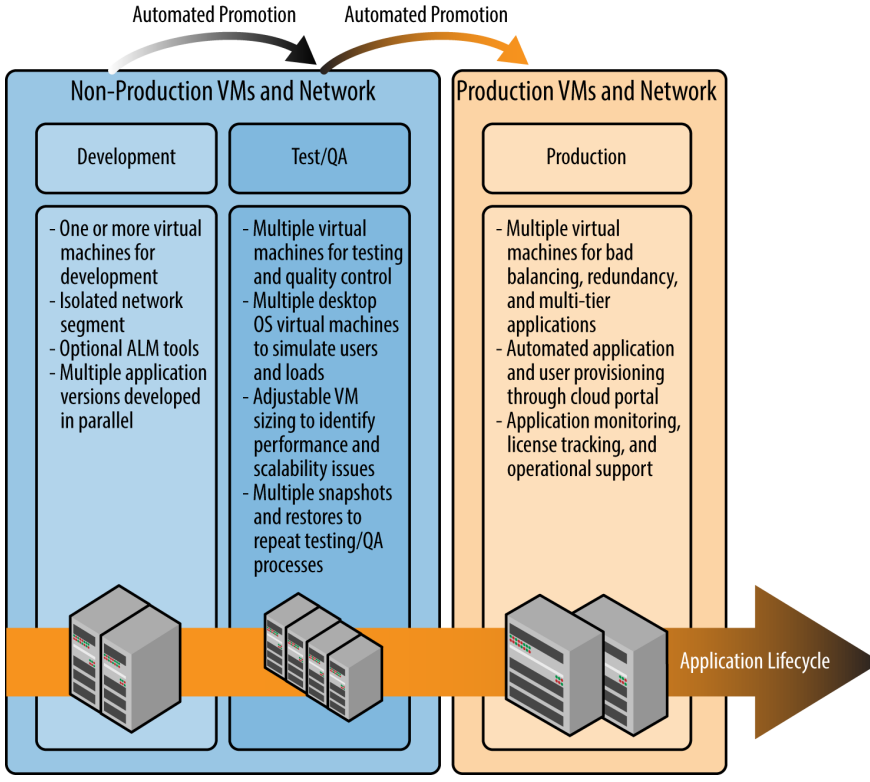

Figure 2-4. Continuous application development and delivery process

Now, consider adding preconfigured VMs with your favorite application lifecycle management (ALM) tools that automatically launch when you order a new Dev/Test subscription. Combine this with automated OS and VM templates, and you can truly appreciate how drastically cloud automation can improve a development team's efficiency. Maybe the best part is that you can easily pause, archive, or terminate your Dev/Test environments when they are not needed and thus stop paying for the service—no servers to disassemble or clear, and your VMs are ready to become active whenever you need them again.

Although the continuous delivery process is nothing new to software teams, the integration of cloud automation and multiple software-defined network segments greatly speeds up and governs an Agile software development lifecycle. This new style of application development has essentially replaced the legacy processes of delivering major software releases in one- or two-year increments. Although the cloud is not necessarily responsible for this shift, it certainly provides further automation and elasticity to facilitate the continuous delivery model.

Application Transformation Methodology

Every modernized datacenter or cloud will provide, at a minimum, the basic VM infrastructure, storage, and network services. When you transform mission-critical applications for use in the cloud, your applications can avail themselves of the unique benefits that the cloud offers. Purchasing or deploying a basic IaaS-focused cloud service is just an initial step in an overall enterprise IT transition to cloud. *It is the application porting or redevelopment that will become the long-term path to complete the transition to cloud.*

Key Take-Away

You must assess each application to determine whether a simple porting is possible or if the application will require a complete redesign to migrate to the cloud.

APPLICATION MODERNIZATION STRATEGIES

There are four types of application modernization strategies to migrate legacy applications in the cloud (see Figure 2-5). You first need to carry out a careful analysis of each legacy application to determine the best method to maximize long-term benefits, taking into account time, costs, and user impact. Some legacy applications are mission critical and unique to your business such that the long-term effort to redesign and recode them is worth the time and cost. Then, there will be relatively simple applications that you can quickly port (i.e., rehost or refactor) to a cloud platform, or even eliminate and replace with a SaaS offering from a cloud provider.

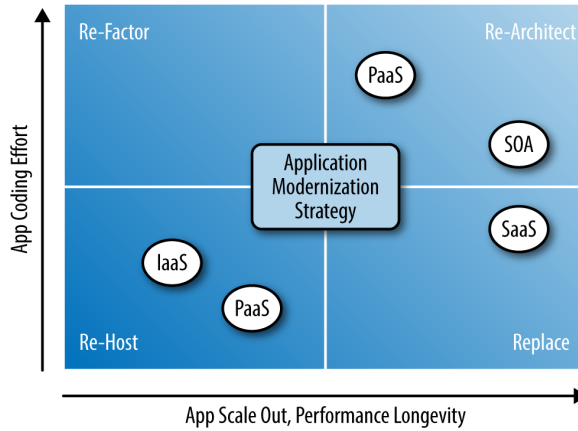

Figure 2-5. Application modernization strategies

Rehosting

Rehosting (or porting) of applications is essentially a "copy and reinstall" technique to take relatively simple legacy applications and host them in the cloud. These applications typically include one or more VMs and, possibly, a database that includes traditional applications which were not originally designed for the cloud. In a perfect scenario, a physical-to-virtual (P2V) migration is possible. Testing and minor configuration changes are often required, but the overall effort to port a single application is usually measured in only days or weeks. Although you might be able to rehost on the cloud quickly and without modifying the application architecture, many of the cloud characteristics (e.g., scalability, resiliency, and elasticity) might not be fully realized.

Refactoring

Refactoring an application is similar to rehosting, but some components of the application are enhanced or changed. You might break up the application into components such as frontend web servers and backend databases so that you can cluster each tier and load balance to provide higher availability, elasticity, and handle more traffic and workload. In this scenario, the core application itself needs little or no reprogramming, because the cloud infrastructure and hypervisors provide some of the scalability without the application having any of the "awareness" that a cloud-native app would.

Redesign

If there are legacy applications that you cannot rehost or refactor (because doing so would not produce the desired performance, quality, or ROI), you should consider redesigning. These applications are often the oldest and most complex legacy applications, possibly mainframe-based, that might require multilayered platforms, databases, middleware, and frontend application servers to be deployed in the cloud. More complex legacy software might require significant assessment and planning just to come up with a project plan and budget to determine the feasibility, risk, and business decision to proceed with the reprogramming. The long-term benefits of a new, properly designed cloud-native application include dynamic elasticity, resiliency, distributed processing, high availability, faster performance, and lower long-term code maintenance.

Replace

Purchasing services from an existing SaaS cloud provider and retiring the legacy application is often a fast and effective cloud-transition strategy. If the ROI to transition a legacy application to the cloud is poor, the organization should consider if the application is truly mission critical or necessary, and whether you could use a COTS or SaaS cloud provider, instead. Technically, you can consider an application hosted in a private cloud as SaaS, but most SaaS applications in the industry are hosted with public cloud providers. Excellent examples of this are public cloud email providers such as Microsoft Office 365 and GoogleApps, or customer relationship management software such as Salesforce.com. Sometimes these SaaS applications provide more features than your legacy software, but they also might not be as configurable as you are accustomed to because they are normally hosted on a shared public cloud infrastructure.

OPERATIONAL CONSIDERATIONS FOR APPLICATIONS

Regardless of which application modernization strategy you use, organizations must also consider operational factors. For example, applications that have been ported to a public cloud might still need database maintenance, code updates, or other routine administrative support. Applications that are hosted as a PaaS or SaaS offering might need less support because the cloud provider will have responsibilities for updating and patching OSs and software components. Of course, in a private-enterprise cloud, your staff (or hired contractor) continue to provide this support but hopefully in a more automated manner than a tradi-

tional datacenter. Consider your current staffing levels, outsourced support, and overall IT organizational structure. Most organizations that have already deployed a private cloud or use some public cloud have not reduced their IT staffing levels; however, they have changed the skillsets and team structures to better accommodate a more service-oriented model that is best suited to support a cloud ecosystem. For more details on recommended operational and support staff changes and best practices, refer to ???.

Application monitoring

When it comes to mission-critical applications that are core to your organization's customers and livelihood, you might keep these applications hosted within a private enterprise cloud or a secure public provider. In either situation, you should still be concerned with monitoring the performance and user experience (UX). The private or public cloud management tools will provide some level of VM, and maybe some limited application-level, utilization monitoring but this is usually not adequate for truly mission-critical applications (they're likely OK for normal business productivity systems). So, regardless of where your mission-critical apps are hosted—public or private cloud—you should still use your own application monitoring tools and techniques that include synthetic transactions, event logging, utilization threshold alerts, and more advanced UX simulated logon tools. For lessons learned and best practices for IT operations, monitoring, and process changes related to cloud-based application hosting, refer to ???.

Service levels

Consider the service-level agreements (SLAs) for applications hosted in the cloud. I cover contracts, terms, and SLAs in ???; however, specific to this chapter's subject, you might need a higher SLA or specific terms for some of your mission-critical applications. Many public cloud providers provide a default level of service guarantee and support that is insufficient for mission-critical applications. In some cases, the public cloud provider does not even guarantee that it will back up, provide credit, or be liable for data loss. Be careful how cloud providers word their SLAs because they might only guarantee network availability in their uptime calculations instead of PaaS or SaaS platform service levels. Other vendors claim extensive routine maintenance windows (in other words, potential outages) that are also excluded from their SLA. Refer to Chapters 2 and 5 for more information on service levels and contractual terms as well as how and what changes to demand from the provider before signing an agreement.

Federated authentication

Consider user authentication and access controls for cloud-hosted applications. You might want to federate an enterprise user directory and authentication system (e.g., Microsoft Active Directory or LDAP) to the cloud for an always up-to-date and consistent user logon experience. As stated earlier, a preferred method is to use a vendor-agnostic industry standard for authentication, such as SAML, especially when federation and SSO is required. For more information about federation and authentication systems, refer to ???.

Scalability

When migrating applications to IaaS or PaaS-based cloud services, you might gain scalability features that were not easy, cheap, or available in the legacy enterprise environment.

Scale out

Depending on the type of application modernization undertaken for a given application, the cloud-based system might now be able to take advantage of dynamic or automated scale out of additional VMs—technically, scale out is called elasticity. It is preferable that the application be cloud native or cloud enabled so that it is capable of detecting peak utilization and triggering scale out automatically. For legacy applications moved to the cloud, you can use the hypervisor and cloud infrastructure to measure utilization with defined thresholds that will trigger scale out, even though the application is unaware of these events. Scaling down after peak utilization subsides is just as important as scaling out. Again, cloud-native applications that handle this automatically are more efficient and faster to react than legacy applications that rely on the hypervisor to scale.

Scale up

Scaling up an application refers to increasing the size of a server, or more common in cloud computing, a VM with more memory and processors to handle increasing workload. Whereas the aforementioned scale out involves launching new VMs to handle peak utilization, scale up involves enlarging the configuration of the same physical server or VM(s) running your applications (up to the maximum number of processors and memory capacity for that particular physical server or VM).

Scale up is considered a legacy technique for scaling. Scale up does not provide cloud-level resiliency or redundancy and is not as efficient compared to scale out. Scale up is considered a legacy technique whose underlying philosophy is "just buy a bigger server" rather than smaller more purpose-built servers and services in a scale-out cloud configuration. Another downside of scale up is that you often need to reboot the VMs to recognize the new processor and memory configuration. However, the need for this additional step will likely recede because some hypervisor platforms are beginning to support dynamic *flexing* of additional processors and memory.

Finally, consider scalability of your applications in terms of geographic access and performance. Although I discuss geographic data sovereignty and redundancy in Chapters 3 and 6, respectively, here I'm referring to load balancing and/or hosting applications in multiple geographic locations to maximize performance and reduce network latency (inherent delays in the communications path over the network). You might want to deploy your application in multiple cloud datacenters on opposite sides of the country in which you reside or in different regions of the world so that end users of your applications are automatically routed to the closest and fastest datacenter. Be aware, however, that many cloud providers charge additional fees for data replication, geo-redundancy, bandwidth, scale up/scale out, and load-balancing capabilities. For an enterprise private cloud, these geo-redundant communication circuits are often cost prohibitive.

Application performance and benchmarking

When you migrate applications to the cloud, you should keep in mind that most public cloud providers do not guarantee the performance of your custom applications nor for any customer-managed PaaS databases or platforms. The cloud provider is simply trying to avoid the argument of who is at fault if an application—particularly one that they didn't create or manage—is not performing the way the customer believes it should. Hosting your own cloud keeps you in complete control of your applications and their performance.

Poor application performance in the cloud is often an indicator of a legacy application ported to the cloud that has not been optimized. For example, just because an application might be copied to a technically faster cloud—"as is" with little or no modifications—does not mean the legacy application will perform well. Performance testing—using live test users and possibly load-testing tools—is recommended for all applications before and after porting them to the cloud.

Having the original application performance baseline measured before any transition to the cloud will give you valuable data to determine expected performance levels. You might be able to use the scale-up or scale-out techniques described earlier to improve performance and meet acceptable levels without redesigning the entire application.

Network bandwidth

Most public cloud providers do not charge for uploading or importing data but do charge transaction, metered bandwidth, or storage input/output fees for network bandwidth. Given that your applications are moving to the cloud for the first time, it is often very difficult to estimate the bandwidth over the Internet or other network circuit. This can result in a bit of a surprise at the end of the first month that the application goes into production use. This bandwidth issue is a much lesser issue for private clouds, because they are usually hosted within your organization's datacenters or via private network circuits. Some public cloud providers offer an optional direct connection option whereby you pay for a private circuit into the provider's network, bypassing most of the normal variable bandwidth fees in lieu of the fixed, direct connect fee. This is well worth it for high utilization/bandwidth needs.

Application Assessment

Assessing or evaluating your existing applications is often the most time-consuming part of the cloud transition. Implementing a private cloud, or procuring some public cloud services, is often completed within one year; however, assessing and then migrating a dozen and up to 100 applications could take a couple of years. The time it takes to perform assessments depends on (at least) three factors:

- Internal technical capabilities versus hiring external application transformation consultants
- Quantity of legacy applications and how many are complex or custom built
- Available budget as it relates to how many parallel assessment teams and work streams

Although hiring application transformation specialists will greatly improve and speed up the assessment and migration planning process, there are steps

that many organizations can take to at least begin their application assessments using internal resources. Using the guidelines and checklists that follow, you might be able to self-assess many of your basic applications—saving you precious time and money—and defer hiring external application transformation-to-cloud experts for only the most complex requirements. If your organization is very large or has hundreds or thousands of applications, seriously consider using these application transformation experts who specialize in *app modernization factories* that have numerous experienced teams and proven processes for analyzing and migrating large quantities of applications in parallel work streams. There really is a special art to doing this type of application transformation in mass, across multiple work streams with the same standards, same governance, and same customer/cloud infrastructure requirements.

Figure 2-6 shows a recommended high-level, five-step application assessment plan.

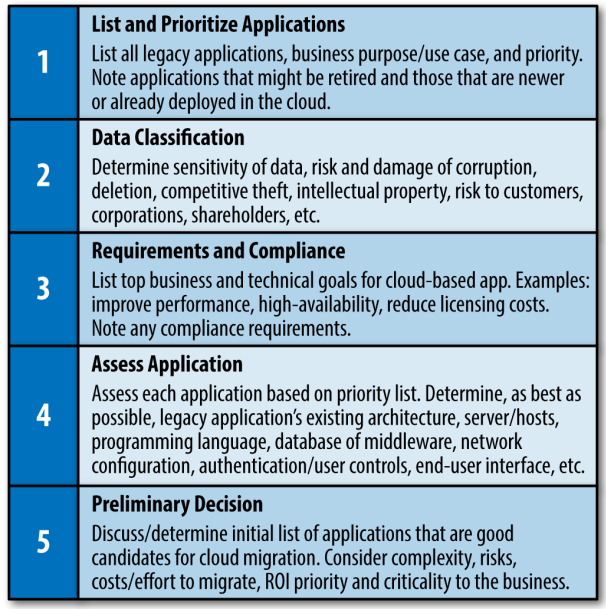

Figure 2-6. Application assessment steps

Table 2-1 contains a checklist of tasks and items to consider during the application assessment process. Even if your organization decides to hire experts for more formal application assessments or actual application migrations, the infor-

mation gathered by completing this self-assessment can be a significant head start.

Table 2-1. Initial application assessment checklist

Category	ASSESSMENT ITEMS/DATA GATHERING
List and prioritize applications	• Create a list of all COTS and custom-built applications along with the primary use case or business function performed by each • Specifically note the name of each application, the manufacturer/software vendor, and version of the application, if known • Note any significant customizations to COTS applications that have been made and any updates that might have been intentionally skipped or avoided due to potential conflicts with these customizations • Prioritize these applications lists based on criticality to the business, how broadly the application is utilized across the business (i.e., how many users), and if this is a customer-facing or internally focused application • Flag applications that are seldom used, are candidates for retirement, have been considered for replacement already, and any workloads for which the cloud has been considered already
Data classification	*Lower Impact Level* The unauthorized disclosure of information might have a limited adverse effect on the organization. *Moderate Impact Level* The unauthorized disclosure of information might have a serious adverse effect on the organization. *High Impact Level* The unauthorized disclosure of information might have a severe or catastrophic adverse effect on the organization. • Consider each application and particularly its data; rank the impact to the organization if the data is corrupted, or completely lost (requiring a data restore); repeat this data

Category	ASSESSMENT ITEMS/DATA GATHERING
	security assessment for all applications and data using the same ranking and criteria • Assess the impact to the internal organization but also to your customers; in addition, weigh the potential harm to your company reputation • Consider the cost and impact of data (e.g., trade secrets) lost to competitors • Consider loss or corruption of customer data, the impact on your customers, and the impact to your organization that this can cause (damage to your reputation, legal issues, monetary damages, and other liabilities) • Classify applications to determine which model they should follow (i.e., the highest-risk applications/data are likely candidates for a private cloud, whereas less-risky applications/data are candidates for public or community cloud models) • Rank each application using one of the following impact categories:
Requirements and compliance	• Briefly list the top business and technical goals (if known) for the application—and possible next generation of the application • Goals or requirements might be to improve application performance problems, reduce licensing purchasing costs, improve user experience/usability, and improve reliability/high-availability • Also note any compliance or similar requirements such as industry or government regulations that might impact the application design, security controls, where data is stored or who has access to and administration of the application and data
Application architecture	• Is the application currently hosted on a single server, spanned across multiple services? Is there a backend database? Can frontend services (web, client-facing application interfaces) be separated into their own network segment from the rest of application, database, middleware? • Does the current application use a multitiered architecture such as separate database, middleware, and frontend processing

Category	ASSESSMENT ITEMS/DATA GATHERING
	services? Can the application and middleware be separated into its own network segment, forming two or three tiers of networks? • Can or should the application share a common database, middleware, or other application or PaaS-type services? Shared services could increase security but reduce licensing costs and easier to manage in an automated environment. • What application platform or programming language was used as the basis for the application (if known)?
Application modernization and migration	• For every application, consider the cost, effort, and risks to redesigning/recoding and if the application is worth the effort, cost, and risk; consider moving commodity applications, such as email, to hosted or even public cloud services • Hire outside consultants and experts in application transformation, if needed, to provide more detailed analysis (even down to code level) if necessary • Which applications could be ported "as is" to the cloud using scaled-up (more computer power) cloud servers? Which applications could be ported to the cloud and use hypervisor-level scale out (such as additional frontend servers) without the application having to be recoded? • Evaluate which applications would benefit from application redesign to take advantage of automation, elasticity, on-demand pay-as-you-go cloud services
Application management	• Always consider how consumers will order applications, how automation will provision them, and how other automated processes will upgrade and monitor them • What application settings, customizations, and self-service controls should be available to administrators and users? Is there a commercial control panel already available on the market or will this need to be programmed as part of the application transformation and deployment in the cloud? Avoid relying on individual app management consoles for each application.

Category	ASSESSMENT ITEMS/DATA GATHERING
	• Will there be billing or financial chargeback of application, data, transactions or data fees to the consumers or other departments? Consider how the cloud management platform will handle this. • What application statistics and reports will need to be presented in the cloud management portal to the customer/consumers? • What user roles and groups need to be created/managed? • How will user authentication and identity for logon to each application be managed and federated through cloud management or other systems? • Consider how you should treat user accounts and data, in each application, when the user no longer exists in the organization, the account is removed, and so on.
Operations and governance	• Evaluate who is currently, and who should in the future, be performing all application upgrades, data maintenance, and monitoring. Are there existing challenges that could be addressed as part of application transformation (change in personnel responsibilities, governance, etc.)? • Consider grouping similar application profiles from an operations standpoint into the same cloud model (i.e., private cloud with operations/management by internal employees) versus applications that are commodities (meaning, they could be operated by anyone internal or outsourced) versus mission critical (meaning, they should only be run/operated by specific persons or department). • Consider advantages and disadvantages of outsourcing application management and upgrade to an external cloud provider (public cloud model) or peer agency (community model); consider which applications are commodities and where an existing SaaS or PaaS cloud provider has a similar or better offering
Mission criticality	• Assess how critical the application—and particularly the data—is to your customers and/or the mission of the organization.

Category	ASSESSMENT ITEMS/DATA GATHERING

This can be a combination of data availability, slow performance, or potential loss of productivity:

Availability	• Low impact to the business if application is unavailable for more than eight hours • No significant financial or measurable productivity impact to employees or customers • Alternative methods exist during extended system outage	• Moderate impact to the business if application is unavailable for more than one hour • Potential financial and productivity losses internally • Potential customer impact including minimal loss of revenue and customer satisfaction • Alternative methods are not adequate during extended outage	• High impact to the business if application is unavailable for more than five minutes • Significant financial and productivity losses internally • Significant customer impact including substantial loss of revenue and damage to reputation • No alternative methods exist during extended outage

Category	ASSESSMENT ITEMS/DATA GATHERING			
	Performance	• Application response time (latency) to user requests is not a concern • Real-time processing of records/ data is not required • Application does not require high availability • Application disaster recovery (DR) to secondary site not required • Recovery point objective (RPO): 24 hours; recovery time objective (RTO):8 hours	• Application response time (latency) to user requests is a concern and must be measured and monitored • Processing of data must be completed as soon as possible but not necessarily in real time • Application must be configured for high availability within single datacenter • Data replication and/or snapshot every 8 hours • Application DR to	• Application response time (latency) is critical with strict monitoring, threshold alerting, and remediation • Real-time processing of data is required • Application must be configured for high availability across multiple datacenters with immediate failover • Data replication in real time is required across datacenters

Category	ASSESSMENT ITEMS/DATA GATHERING
	secondary site required • RPO: 8 hours; RTO: 4 hours
Preliminary decisions	• Form an initial decision as to which applications are good candidates for a cloud migration, which apps should remain hosted within internal datacenters, and which workloads should not be migrated or dealt with immediately • Consider which applications and workloads are best fits for hosting within an internal private cloud and which might be appropriate for a public cloud • This preliminary decision should be based on the assessment steps described earlier compared to the effort (cost, time, ROI) that the migration will require and ultimately the priority to the corporation • Consider hiring external application transformation-to-cloud consulting services to handle the most complex or mission-critical workloads. Have them perform detailed assessments and systems redesigns, select cloud providers/models, develop a cloud migration plan, and conduct a pilot.

Category	ASSESSMENT ITEMS/DATA GATHERING
	• Most organizations have numerous applications and business priorities, so aligning these is crucial to forming a realistic cloud migration plan that meets available budgets and timelines. A common approach is to "continuously reprioritize" the application migration efforts over time to keep up with evolving business priorities.

Application Transformation Best Practices

Based on lessons learned and experience from across the cloud industry, you should consider the following best practices for your organization's planning.

LEGACY APPLICATION ASSESSMENT

Assessing each legacy application is an essential part of your cloud planning and transition strategy. Use the following guidance when evaluating each of your existing applications:

- Analyze each application to determine which architectures, multitiered applications, or legacy applications you could move quickly to a cloud (public or private) and which will require more significant transformation.
- Consider data security and risks on an application basis. Are there applications and data that would be at risk if hosted by a cloud provider or possibly in another state, territory, or country?
- Consider breaking up legacy applications into multitiered platforms as part of the transition to cloud. For example, separating application data and databases from middleware and frontend application servers will allow more elasticity, reliability, scalability, and possibly an ability to use the data platform by other applications that are also transited to the cloud. In this analysis, consider which applications you can transform and have share a common platform rather than moving every legacy workload to the cloud as individual applications.
- Remember, you can always leave an application back in the legacy/enterprise datacenter and deal with it another day. Some organizations and businesses need to show a more immediate benefit and adherence to cloud-first standards so don't necessarily take on the difficult applications first.
- Application assessment checklist:

  List and prioritize applications

  List all legacy applications, their business purpose and use cases, and priority to the business. List applications that might be retired or are seldom used.

Data classification
: Determine the sensitivity of the data for each application. Assess the risk of data corruption and competitive theft of intellectual property compared to the harm this might cause the corporation, your customers, or shareholders.

Requirements and compliance
: List your top business priorities and technical goals for each application. Do legacy applications have performance problems or require change regardless of the cloud migration? Note any regulatory or security compliance requirements.

Assess applications
: Assess each application based on the priority list. Determine as best you can the legacy application's software architecture, servers/hosts, programming language, database or middleware, network configuration, authentication/user controls, end-user interfaces, and so on.

Preliminary decision
: Discuss and determine your initial list of applications that are good candidates for cloud migration. Consider application complexity, risks, costs/effort to migration, ROI and priority and criticality to the business.

APPLICATION MODERNIZATION TECHNIQUES

Evaluate each legacy application to determine if, when, and in what priority to migrate the system to the cloud. Based on the assessment, select from the four application modernization strategies:

Replace

Depending on your business priorities, it might not be cost effective to recreate some legacy applications for the cloud, so consider porting these "as is" to a cloud provider or replacing the legacy application entirely with a new public-provider hosted SaaS offering.

Rehost

It might be possible to copy and reinstall less complex applications in a cloud environment with little or no changes. Testing and network address changes are often required.

Refactor

You can redeploy multilevel applications into the cloud using multiple VMs to gain more performance, reliability, or scalability.

Redesign

When legacy applications are critical to the business and you cannot use the other migration techniques, a redesign and reprogramming of the software might be required. Although the advantages of the new modern application in a cloud are numerous, you need to make a financial decision to determine if the effort and cost are worth such investment.

CONSIDER CLOUD ARCHITECTURES FOR NEW APPLICATIONS

You should consider a cloud-based design and operations approach for all new applications and IT systems to achieve scalability, elasticity, and resiliency. Do not forget that the business outcomes and consumers of the applications is also critical. Here are some considerations:

- You should build applications with embedded multithreading, multitenant, highly scalable architectures to span across multiple servers, VMs, datacenters, and cloud providers.
- Though new applications can start on a small infrastructure, having these inherent capabilities will greatly improve the ability to make applications redundant, resilient, and scalable—all part of reduced operational, management, and support costs in the long term.
- Implement new application development practices around continuous development and delivery.
- Consider cloud-native application characteristics in every new application —and as a goal for all applications that are being redesigned as part of your cloud migration.

OPERATIONAL CONSIDERATIONS

Consider the following recommendations when evaluating your legacy applications and transitioning them to a new cloud operating environment. Remember that simply moving or porting applications to the cloud without modification is certainly possible, but may not provide the best performance, scalability, or long-term supportability.

- Consider establishing unique service levels for mission-critical applications rather than accepting the default cloud-wide service level proposed by the cloud provider.
- Implement federation tools to connect your enterprise user directory and authentication system to any public or managed private clouds to provide SSO capabilities to your users. This also greatly helps maintain security and permissions by having an always up-to-date user directory.
- Use scale-out and scale-up techniques to increase or decrease system capacity and application performance as applications workloads change. You might be able to use these scaling techniques to improve migrated legacy application to achieve better performance even if the app was not rewritten specifically for cloud hosting.
- Measure application performance of your existing enterprise applications before and after you migrate or port them to the cloud. This will make it possible for you to properly set scaling options and avoid the "blame game" with the cloud provider if application performance problems are found.
- As part of testing and during the initial days, weeks, and month of a new application hosted in the cloud, pay careful attention to network bandwidth utilization so that you are not surprised when the end-of-month invoice is calculated. Remember that most public cloud providers charge for network bandwidth (sometimes after a base allowance is exceeded).

REPLACE COMPONENTS AND LEGACY LICENSING AGREEMENTS

While evaluating your legacy applications and determining whether or when to move them to the cloud; consider renegotiating any software licenses, replace components of the system with COTS software, or use cloud-based platform/PaaS offerings. Here are some considerations:

- As part of the migration to cloud, consider replacing certain components of the system with COTS software of a PaaS offering from the cloud provider.
- Consider changing or renegotiating a legacy software license with a different vendor or in a more pay-as-you-go model—chances are that the legacy software platform and license agreements haven't kept up with modern licensing practices or pricing models.
- Consider what commercially available SaaS offerings are available in the industry and whether your organization could save money on internal software development, maintenance, and hosting. Many SaaS offerings might even provide more features than your current applications and might be able to assist in data importing/transition.

About the Author

James Bond has more than 25 years' experience in the IT industry and has designed and deployed countless datacenters, server farms, networks, and enterprise applications for large commercial and public sector government clients—he was building hosted application services long before the term "cloud" was first used in the industry. Mr. Bond is a business and technical cloud subject matter expert, providing cloud strategy, guidance, and implementation planning to C-level executives seeking to transition from legacy enterprise IT to cloud computing.

Mr. Bond currently works for Hewlett-Packard as a cloud chief technologist. He routinely presents executive briefings at industry conferences and in-depth consulting workshops on lessons learned to large commercial and government organizations. His specialties are enterprise IT transformation to private and hybrid cloud as well as cloud brokering. Prior to Hewlett-Packard, Mr. Bond built numerous cloud computing companies and practices serving in the roles of chief technology officer, product vice president, chief architect, and software development management.

Mr. Bond has a bachelor's degree in information technology from the University of Maryland and has received numerous industry certifications and awards throughout his career. He is a well-respected industry leader and longtime contributor to numerous trade magazines and a featured speaker at IT conferences. This is his first published book.