Distributed Systems: A Modern Approach

Prashant Shenoy

University of Massachusetts Amherst

1. Cloud Computing

With the advent of the Internet, cloud computing has become the most common method to run distributed applications. Cloud computing involves the use a network of remote servers to run distributed applications. The Oxford dictionary defines cloud computing as "the practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or personal computer.''

Application Deployment in the Pre-Cloud Era

To understand why cloud computing has become so successful, let us examine the traditional method of running distributed applications in the pre-cloud era. This involved not only developing the application but also establishing the computational infrastructure needed to deploy and run the application. Consider the ACME Toy Company that wishes to set up an online web-based toy store to sell its plush toys over the Internet. While its software engineers are busy developing its online store application, its engineers also need to purchase an appropriate number of servers and storage to run the application. By appropriate, we mean an adequate number of servers to handle the peak load seen by its website, which is the maximum number of customers that access its online store concurrently. These servers will need to be deployed in a server room, also known as a data center. The data center will need to be equipped with cooling equipment for the servers and an Internet connection to provide network connectivity. Needless to say, building its own compute infrastructure is expensive and laborious, and the process can take several months or more. In addition, ACME needs to address several challenges when designing the compute infrastructure for its online store.

- *Capacity planning.* Before launching its online toy store, ACME must estimate the peak load that will be experienced by its web-based store in order to determine how many servers to purchase. This is a challenging task especially since ACME may only has a rough estimate of how many plush toys it can expect to sell in its online store. It is easy to make mistakes by over-estimating or under-estimating demand. If its latest plush toys become a run-away hit, and ACME underestimated their popularity, its online store may get overloaded and its website may crash or turn away requests from online customers. On the other hand, if its toys do not sell as well as its expected, it may be left with servers that are highly underutilized. Moreover, since the demand seen by ACME's online store may experience gradual growth as its online sales increase over time. Capacity planning, which involves determining how much server capacity to provision for its online store, is challenging due to ACME's inability to accurately predict future demand. Adding new servers can take weeks or months if demand is underestimated, which can lead to lost sales and poor user experience on an overloaded website.

- *Low resource utilization.* If capacity planning is done properly, ACME's compute infrastructure will have adequate server capacity to handle the peak workload seen by

its online store. Most days will however see an average workload that is substantially lower than the peak workload, which is seen only during the busiest shopping days of the year. As a results, its servers will experience low average utilization or idle capacity that is wasted on most days. This is a common problem for many web applications that experience a high peak to average workload ratio; resources are underutilized or wasted whenever the mean workload is significantly lower than the peak (see Figure X).

- *Lack of agility.* In contrast to low average server utilization during typical days, ACME's servers will experience a heavy workload and high utilization on busy online shopping days such as Thanksgiving Black Friday in the United States and Single Day in Asia. They may also experience a sudden, very large workload spikes, which is referred to as a flash crowd. Flash crowds can occur during a flash sale of ACME's popular must-have plush toy of the season. News websites often experience a flash crowd during very significant news events. Peak workloads or flash crowds can result in server overload, especially if the actual peak workload exceeds the estimated peak workload used during capacity planing. Since computing infrastructure is provisioned by ACME a priori, it has limited ability to add new servers at short notice to its online store. Thus, it suffers from a lack of agility to scale the capacity of its online store to respond to sudden or unexpected workload spikes from flash crowds or higher than expected peak workloads on popular shopping days.

- *Economy of scale.* Computing infrastructure deployments benefit from economies of scale as they grow larger—since the cost of building a data center can be amortized across a larger number of servers. Smaller data centers are more expensive to build and maintain on a per-server cost basis. If ACME's compute infrastructure is designed to support a single distributed application, namely its online store, it will be unable to reap economy of scale benefits that accrue from larger-scale deployments.

Hosting platforms and Utility Computing.

Web hosting platforms emerged in the 1990s to address the above challenges faced by application providers when building their own infrastructure. A web hosting provider deploys a large compute infrastructure in their data center and offers a hosting service, which is the ability for application providers to deploy and run their applications on remote servers that belong to the hosting platform. Since a hosting provider can host a large number of applications from a larger set of customers on its platform, the economy of scale incurred from its large data center translates to lower costs for its customers. In the model, ACME engineers no longer need to build their own computer infrastructure and they can can focus solely on developing its online store. Instead ACME rents the necessary server capacity from the hosting provider, in return for a monthly server rental fee, and uploads its online store application code and data to these servers for remote hosting. With the advent of the Internet, many hosting platforms mushroomed in the late 1990s and the early 2000s. As it became easier to access and use remote computing resources, distributed systems researchers envisioned that computing would become soon a utility, where computing resources would be available on-demand over a network to anyone who needed them, much like electricity from an electric socket. This model was referred to as utility computing.

Cloud Computing Benefits.

Today's cloud computing is the modern realization of this utility computing vision and the early web hosting services. As we will see, modern cloud platforms offer many more services than simple web hosting. But servers (or more precisely, virtualized servers) continue to remain a basic building of computation offered by modern cloud computing platforms. Like web hosting platforms, cloud platforms offer computation and storage resources, as well as higher-level services, for lease by application providers. They also offer more flexibility on how resources are acquired and billed. A cloud customer can request a single server or hundreds of servers, and these resources are provisioned and ready for use in a matter of minutes. The customer can use these server resources for as long as needed and relinquish them by shutting them down. Server usage is billed on a per-hour, or per-minute, basis. The advantages offered by cloud computing are as follows.

- *On-demand resource allocation.* A key advantage for application providers is that they can request servers for their application at any time and the cloud platform allocates the requested servers from its pool of idle servers in an near instantaneous manner. Compare this to the pre-cloud approach where new severs need to be purchased and installed, a process that can take weeks or months. This flexibility can increase the speed of deploying new applications or making changes to current applications.

- *Elastic scaling.* A direct consequence of the on-demand allocation model is the ability to easily scale the server capacity provisioned for an application up or down. This is referred to as elastic scaling. If the workload seen by a web application starts rising, additional servers can be quickly acquired from the cloud platform and added to the pool of servers executing the web application. If the workload drops and servers become underutilized or idle, some of them can be relinquished. Elastic scaling simplifies the process of provisioning additional servers during peak periods or to handle unexpected flash crowds.

- *Pay as you use Model.* On demand allocation also comes with the benefit of no upfront commitment for a certain number of servers, since servers can be acquired and relinquished as and when needed. Cloud computing platforms use a per-minute or per-hour billing model, where each server is billed based on the number of minutes or hours for which is it held by a cloud customer. For example, if a server is acquired and then released after two hours, the customer only pays for two hours of usage. The ability to use resources for short time periods reduces cost for the application provider. Conversely, if the customer is willing to make long term commitments for servers they need for longer time period, the cloud platform offers discounted pricing for such commitments.

- *Economy of scale.* Since cloud computing providers server thousands or even hundreds of thousands of customers of all sizes, ranging from individual developers to large enterprises, they can build very large data centers that house hundreds of thousands of servers in a single data center. To serve a global clientele, these data centers are distributed in many different continents and countries. The economy of scale resulting from these large deployments as well as hardware improvements from Moore's Law translate to lower per-server costs, and lower prices for customers. In fact, the price of cloud computing servers has fallen steadily over the years. For example, the cost of a small server on a cloud platform has fallen from 10 cents per hour in 20XX to 1.7 cents per hour in 2020, a factor of 5 decrease!

Now consider how ACME can build its online toy store using a cloud computing platform. It no longer needs to spend months building its own compute infrastructure and can instead provision a cluster of servers from the cloud platform in minutes. Capacity planning is still necessary to determine the number of servers to provision in the cloud, but it is much easier to deal with errors in its estimated peak workload. If it underestimates the peak workload and demand ends up higher than expected, this is easily addressed by acquiring more servers from the cloud platform. Overestimates can be handled by relinquishing unneeded servers. Low resource utilization can be handled by using a dynamic pool of servers, where fewer servers are used during periods of low utilization. Adjustments to server capacity can be made frequently (e.g., daily or weekly) or infrequently (e.g., monthly) depending on its desire to balance cost savings with reconfigurations of it server resources. Peak workloads and flash crowds can be handled by rapidly provisioning additional servers and elastically scaling up the capacity of its online store. In other words, ACME's engineers can focus on building a high quality online web application and exploit the above benefits of the cloud computing platforms. This flexibility does come at the cost of an increase in application complexity, where the online store application needs to be designed with the ability to handle, and even drive, changes in server capacity. As we will see in subsequent chapters, cloud platforms support many possible features to simplify such elastic scaling.

Cloud Delivery Models

[Need a better way to introduce the notion of cloud delivery model] While ability to rent a server remain a basic building block of the service offered by cloud platforms, cloud services can be delivered in many different ways. There are three different delivery models for offering cloud services.

Infrastructure-as-a-Service (IaaS): In the infrastructure-as-a-service model, the cloud platform provides computing resources in the form of servers or disk disk storage to its customers. The application provider can use these resources as building blocks to construct their own compute infrastructure in the cloud. For example, the application provider can lease a certain number of servers and a certain amount of storage and then manage this pool of leases resources as per their needs. The cloud essentially provides barebone servers and storage and leaves all aspects of managing these resources to the the application provider. The application provider needs to install the desired OS and software components and then deploy the application code and its data on these resources. The IaaS model is akin to acquiring hardware servers and storage in the pre-cloud era and then configuring them as needed for subsequent use. Modern platforms offer dozens of different server configurations. In the early years, these cloud server configuration differed the the number of CPU cores, amount of memory and local disk and network capabilities. More recently, cloud platforms also offer a choice of CPU type (e.g., Intel x86 or ARM processor), CPU clock speed, type of local storage (e.g., solid state disk or traditional hard disk), and special hardware such as GPU. Storage offering are similarly varied and support data storage of online, nearline, and archival data, as discussed below. These plethora of server and storage configurations offer significant flexibility to an application developer but also complicate decision making since they can choose from hundreds of different sever configurations. As we will discuss below, server resources are charged by the amount of time for which each server is used by the customer; usage is billed by the hour, or in some cases, by the minute. Storage resources are billing by the amount of stor-

age space, the volume of reads and writes, and network bandwidth used for these I/O operations.

Platform-as-a-Service (PaaS): In the Platform-as-a-Service model, the cloud resources are offered in the form of a runtime platform that can be used by application providers to run their applications. In the case, the application providers upload their application code to this platform, which then takes care of deploying the code on servers and provisioning adequate capacity to each deployed applications. From the application providers perspective, they no longer need to worry about estimating the server capacity needed their application, acquiring those cloud servers, and managing them on an on-going basis. The cloud software platform provider is responsible for these decisions for each deployed application. The platform also takes care of scaling the capacity allocated to an application whenever it sees a workload increase. In our example, ACME no longer has to acquire and manage cloud servers for its online toy store. It can use a software platform designed to run web applications written in popular languages such as Python, Java, .NET, or Node.JS and upload code and data for the application to the platform. The platform takes care of provisioning servers, deploying the application on these servers, and managing these servers on behalf of the application providers. The PaaS model is suitable not only for running full-fledged web applications but also for running the backend component of mobile applications and various types of serverless computing workloads (see Ch. XX).

Software-as-a-Service (SaaS): In the Software-as-a-Service model, cloud resources are offered in the form of packaged applications. The SaaS model has become a common method for selling software applications by software companies. The advantage of the model is that end-customer does not need to deploy any software packages on their local machines, and they connect to a cloud server running the packaged application, often through a web browser or a lightweight mobile application in case of mobile devices. SaaS applications are also becoming the preferred method to sell software since billing is based on a recurring subscription model, rather than a one-time software purchase, which ensures a steady revenue stream for the software company. Examples of common SaaS applications includes web-mail services such as Google Mail, Google docs, Office 365 online, among others. SaaS cloud platforms such as Shopify even provide a full online store as a SaaS offering. ACME can choose to use such a SaaS offering to quickly deploy their online toy store. To do so, it needs to upload its catalog of toy product to the online store SaaS platform and customize the look and feel of their of the online toy store—little to no code development is required to build and deploy their store.

It is worthwhile comparing the three delivery models from an application providers standpoint. In the IaaS model, the application provider needs to explicitly lease servers and storage from the cloud provider and deploy their own application code and data on these resources. The application provider is responsible for managing their servers, storage and their application deployment. In the PaaS mode, the application provider develops and deploys their application code onto a cloud-provided software runtime platform. The platform is responsible for managing servers and storage for each deployed application and the application provider need not worry about these lower-level hardware and software infrastructure components. Finally, in the SaaS mode, the cloud platform provides the full hardware and software stack consisting of servers and storage resources, OS and runtime software, and the application software. The customer no longer needs to lease infrastructure resources or even develop their own application software and simply uses fully-built applications provided by the SaaS cloud platform.

Deployment Models

Cloud computing platforms can be designed for general use by any customer who is willing to pay for these services or for private use by a single customer, usually an enterprise company. Depending on how the platform is built and who uses it, cloud deployments can be classified as public, private, or hybrid.

Public Clouds

A public cloud is a cloud computing platform built by a cloud company (referred to as a cloud provider) that is designed for general use by the public at large. Any application provider (i.e., cloud customer) can sign up for cloud service from a public cloud and request computing and storage resources. A public cloud is designed to provide service to a large number of customers. Hence, compute and storage resources are shared (multiplexed) between a number of customers and their applications. The public cloud platform must isolate customers from one another for performance and security reasons. An application belonging to one customer should not interference with, and impact the performance of, an application belonging to a different customer. From a security standpoint, data and applications belonging to a customer should be not be accessible to other customers. Such isolation is provided through OS and network resource management mechanisms such as virtualization (Sec XX). Such isolation mechanisms also provide transparency, which means that a customer only sees their applications and data and the presence of other customer's applications is not explicitly visible. Major cloud providers such as Amazon Web Services (AWS), Microsoft Azure, Google Cloud and Alibaba Cloud are examples of popular public cloud platforms. Each of these public cloud platforms supports tens of thousands of customers of all sizes from single person businesses to some of the largest businesses. To do so, these cloud platforms operate on a global scale by deploying large data centers in multiple countries and continents, allowing them to service each customer using a data center in their local country or a nearby one.

Private Clouds

A private cloud is a cloud computing platform designed for a single customer, usually an enterprise business. A large enterprise may choose to build its own private cloud instead of using a public cloud and offer cloud services to its employees. Similar to a public cloud, employees can request computing and storage services on-demand from the private cloud. In this case, the private cloud multiplexes it resources across applications and data belonging multiple employees, similar to how a public cloud does so across multiple third-party customers. The main advantage of a private cloud is that servers and storage belonging to the private cloud reside on the company's internal network and behind the network firewall. The enterprise not only has full control over its servers and storage (since it owns the hardware) but also its application and data stay "inside'' the company's network, which can potentially be more secure. Typically private clouds have been built by enterprises by deploying their own compute infrastructure on their internal networks for their private use.

More recently, public cloud provider have begun to build and deploy private clouds for their large customer's internal use. Doing so allows them to utilize their expertise in efficiently managing public clouds for the purpose of building and operating private clouds for enterprise customers. As an example, in the United States, Amazon Web Services oper-

ates GovCloud, a large private cloud for use by the US Government and its many branches [Note: IS GovCloud really a private cloud or a highly secure public cloud with restricted access to govt entities?]

Hybrid Clouds

A hybrid cloud uses a mixes of private and public cloud resources. A hybrid cloud can be viewed as an extension of a private cloud where the private cloud is augmented with public cloud resources. To understand why this may be necessary, let us consider the overall demand for servers seen by a private cloud from its internal users. Like a public cloud, this demand will vary over time and may see sudden spikes from time to time. For example, at the end of a financial year, several departments within ACME ranging from sales to engineering may require additional serve capacity to handle end of the year needs. ACME's online store may also experience higher seasonal demand due to the holiday season. As additional servers are requested from the private cloud by multiple users concurrently, the capacity of the private cloud may be occasionally exhausted if the demand spike is high. Rather than denying requests for additional servers when the cloud is "full,'' the private cloud may choose to temporarily acquire additional servers from a public cloud and add it to its pool of cloud servers. Network mechanisms such as virtual private networks (VPNs) are used to make these public cloud resources appear as virtual private cloud (VPC) resources, seamlessly making them part of the private cloud. This also unifies two pools of cloud resources—from the private and public cloud–into a single hybrid cloud pool.

Hybrid cloud offer some advantages and disadvantages over a private cloud. A key advantage is that it allows an enterprise to build a private cloud without worrying about the need to handle unpredictable demand spikes. It can purchase fewer servers for its private cloud, and reduce the number of idle or underutilized servers that remain on standby for peak demand periods. By temporarily acquiring public cloud servers during periods of peak demand, it can still handle its user's needs and avoid having to invest in a larger pool of servers for its private cloud. However, in doing so some of its hybrid cloud servers are no longer inside its internal enterprise network. They are public cloud servers connected to its internal network over the Internet using VPNs. Some of its data will reside on public cloud servers that are part of its hybrid cloud.

As an example, let us assume ACME has provisioned 100 servers in its private cloud since its overall demand for servers remains under 100 servers for the majority of the year. However, for a few weeks each year, the demand rises to 125 servers, which it satisfies by acquiring 25 servers from a public cloud and creating a hybrid cloud. This allows its private cloud applications to "overflow'' into the public cloud during peak demand periods, an approach known as cloud bursting. Once the peak demand period ebbs, the public cloud servers can be relinquished and returned back to the public cloud. Clearly, it is more cost effective to rent these 25 servers for a few weeks of the year from the public cloud rather than buying these servers and having them remain idle in its private cloud for much of the year.

Cloud Economics

Cloud resources such as servers and storage are offered under a variety of pricing models. These pricing models vary in terms of length of commitment, availability guarantees, flexibility, and cost. As cloud platforms have evolved, these pricing models have also evolved

in terms of choices they offer. Since cloud monthly bills can add up quickly for large customers, it is important to understand these pricing models and their tradeoffs in order to optimize monthly costs.

To understand these choices, let us start with an example. Consider an ACME engineer looking to rent an apartment. A nearby apartment complex offers multiple type of rental leases. A month to month lease requires no upfront commitment and provides the flexibility of vacating the apartment at any time. An annual lease requires a year long commitment from the renter, and is less expensive than a month to month lease. What option should the engineer choose? Obviously a month-to-month lease makes sense if future is unpredictable or if the need is short term (e.g., if the engineer is a summer intern and wants to rent for a few months). An annual lease is a better choice if the engineer's rental needs are long-term, since it also provides lower rental costs. Finally, if the apartment complex has many vacant apartment, it may offer some of them for for short term rentals at steep discounts, with the understanding that it can ask the renter to vacate the apartment with a short notice if it finds a rental customer who is willing to rent the apartment.

- *On-demand Instances.* On-demand pricing is the default pricing model for cloud servers and offers pay-as-you-go pricing. On-demand server instances can be acquired and relinquished at will by a cloud customer. This is the most flexible pricing model since it is required no a priori commitment from the customer on the duration of time for which a server will be held. Customers can request any number of servers (from a single servers to hundreds) at any time and relinquish them when they are no longer needed. Server usage is billed either by the hour or by the minute, depending on the cloud provider. Thus, this model is akin to a month-to-month lease for an apartment, except that the cloud servers can be leased by the hour or by the minute. This flexibility of on-demand allocation and no a prior commitment comes at a cost—on-demand pricing is also more expensive than other types of pricing models.

- *Reserved Instances.* Reserved pricing allows customer with stable resource needs can lease cloud servers for longer durations such as a year or three years. Reserved instances require an commitment to pay for these servers (or storage) for that entire rental duration. In return for this longer-term commitment, they offer substantial discounts over on-demand pricing. The hourly prices of reserved instances can be 40% lower for an annual commitment and up to 60% lower for three year commitments. Reserved pricing is akin to annual leases for an apartment. Similar to apartment leases where the renter has to pay a termination fee for reneging on a lease, some cloud platforms allow customers to relinquish a reserved server instance by paying an early termination fees. Cloud platforms also allow customers to resell their server to other customers through a cloud marketplace and have the new customer take over the remainder of the reserved server lease, which is similar to sub-leasing an apartment to a new renter, who takes over the remaining duration of the lease.

- *Revocable or Spot Instances.* Cloud platforms also offer a third instance type called revocable or spot instances. Revocable or spot servers have an interesting pricing model. They are offered at substantial discounts to on-demand or reserved server pricing (the discounts can range from 70 to 90%). The cloud provider has the right to unilaterally reclaim these servers from a customer and assign them to a different customer. Revocable or spot servers represent the unused or surplus capacity on the cloud platform. Normally, such servers would stay idle and the cloud provider

would not generate any revenue from idle servers. By offering them at highly discounted prices, they can offer them to price sensitive customers, with the option of reclaiming (i.e. revoking) them from these customers in the event of new requests for on-demand or reserved servers. By offering surplus capacity in the form of revocable instances, cloud providers can generate some revenue from servers that would otherwise remain idle. Customers benefit from the ability to lease such severs for short time periods at low cost.

But why do cloud platforms have large surplus capacity on their platforms? This is because cloud providers must themselves provision their cloud platforms to meet peak demand from their pool of customers. The demand on a typical ("average") day is substantially lower than the peak demand, which means the cloud platform will have a significant portion of its servers still idle, allowing the cloud provider to offer them in the form of revocable servers.

While any cloud providers that offer revocable servers offers them at large discounts, the pricing model varies by cloud provider. Some providers offer fixed pricing, or a fixed discount, over the on-demand price for that servers. Other cloud providers offer variable pricing, where the discount itself (and the price) can vary over time depending on demand for those servers. When the demand is low, the price is lowered by offering higher discounts. When the demand rise, the price is gradually increased by lowering the discount. In case of fixed pricing, some cloud providers also specify an upper limit, such as 24 hours, on how long a customer can hold a revocable server. The server is automatically revoked at the end of this maximum holding duration, and it can also be revoked earlier if the cloud platform sees demand from on-demand or reserved server customers.

Due to their revocable nature, such servers are suitable only for certain type of applications. In particular, they are well suited for disruption tolerant applications, which are applications that can withstand disruptions in server availability. An application such as a web server would not suitable for execution a revocable server, since the website would experience a downtime if the server is revoked. In contrast, batch applications are well suited for these type of servers, since the batch job can be restarted on another server upon a revocation event. In this case, revocations only increase completion times and do not impact correctness. For long running batch tasks, it is also desirable to periodically checkpoint the work done thus far to disk. The job can be restarted from the most recent checkpoint, rather than from the beginning, which reduces wasting CPU cycles and avoids long delays in completion of the task.

Revocable servers also have lower chances of being available. If one is requested and the demand for on-demand servers is high, the cloud platform can deny the request. The chances of a denial tends to be very low for other types of servers such as on-demand and reserved. If request for revocable servers is denied, a customer can wait and resubmit the request after a period of time, hoping that some surplus capacity has become available. Alternatively, the customer can request a different configuration of a revocable sever or request one from a different geographic region. Each configuration forms its own market for revocable servers, independent from others, and the chances of a request being denied will vary from one server configuration to another.

Case Study. Cloud Economics and Choosing a Pricing Model for Cloud Servers.

Cloud costs can be optimized by choosing the right type, or the right mix, of pricing models for servers leased from the cloud. The choice of a pricing model also depends on the predictability of computational demand and the dynamics exhibited by it. If a cloud customer does not have good estimates of computational demand for cloud servers, (i.e., experiences variable demand that is unpredictable), on-demand servers are the best choice. They can be requested at will and require no a priori commitment on the duration for which they will be leased by the customer. At the other extreme, if a cloud customer sees a stable and long-term need for cloud servers, reserved servers are a good choice since they offer significantly discounts in return for making a long-term rental commitment.

Next consider a customer who sees a time-varying future demand that can be estimated using some means (such as using past historical demand to estimate future needs). Figure X shows an example computational demand seen by a customer over a one year period. The demand varies by the month and is higher in certain months of the year than others. One option is to satisfy this demand using only on-demand servers, by acquiring additional servers when the demand rises and relinquishing them when the demand falls. Another option is to use only reserved servers to satisfy this demand. Since reserved servers require a year-long commitment, this entails leasing all the servers at the beginning and keeping the number of servers fixed for the entire duration. Since demand varies over time, this essentially requires leasing enough reserved servers to handle the maximum demand. Some server capacity will remain idle and be wasted at other times. The figure shows the mean number of on-demand servers leased over the course of a year will be strictly smaller than the number of reserved servers, but the reserved servers are also cheaper. The cheaper option in this case depends on the percentage discount d of reserved servers. Since reserved servers are d% cheaper, up to d% of the CPU capacity can be wasted and the option will still be cheaper than using on-demand servers. Put another way, reserved servers will be cheaper than on-demand servers so long as the mean utilization (the area under the shaded curve / the total area under the dark curve) is at least (100-d)% where d is the percentage discount over reserved servers.

But there is a third option, which is to pick a mix of reserved and on-demand servers. The customer can use reserved servers to satisfy the base (minimum) demand that is always present and satisfy the remaining time varying demand than on-demand servers. This option is strictly cheaper than using on-demand servers alone, since some servers are cheaper reserved servers. In fact, the optimal solution is to choose enough reserved servers where the mean utilization of those servers equals (100-d)% and to satisfy the remaining demand using on-demand servers. As can be seen, optimizing cloud costs is not an easy task. Our analysis assumes that we can estimate long-term demand well for up to a year in advance. This may not always be possible. But any portion of demand that can be estimated in advance can be optimized in this manner.

Spot servers also offer a means to saving cloud costs. They are typically used for short-term jobs due to their revocable nature, rather than to handle long-term demand. Some researchers have proposed using a mix of non-revocable on-demand servers and revocable servers even for applications such as web servers. Even if spot servers are revoked, the web site does not experience a downtime since on-demand servers can continue to serve incoming requests, while the revoked servers are replaced with new spot or on-demand servers.

Cloud Storage

Cloud platforms offer storage resources to their users, in addition to their server offerings. The ability to store one's data in the cloud, where it can be accessed from any device over the Internet, is attractive to end-users due to its convenience. As a result, the average Internet or smartphone user is likely to be using cloud-based storage to store some of their data. Two specific trends have made cloud storage popular in recent years.

Media storage in the cloud: As smartphones have become ubiquitous in our lives, the use of smartphone cameras to capture photos or videos of our daily lives has increased at an unprecedented rate. Mobile devices such as phone have a limited amount of on-board storage, and cloud storage offer a convenient means to store a lifetime of phones and videos. Services such as Google photos, Amazon Cloud drive and iCloud storage offer the ability to upload photos and video from a smartphone to a cloud storage drive and view them later from any device.

Document storage in the cloud: Cloud storage has also become popular for storing documents remotely, which enables convenient access from any of the user's computers as well as the ability to share documents for collaborative editing. Example of cloud storage services for storing documents or any type of file include Google drive, Microsoft's OneDrive and Dropbox. Google's online suite of document editing applications was an early service that made document editing using a web browser convenient, along with the ability to store all documents on a cloud storage drive. Cloud storage services offer the ability to synchronize files stored in the cloud to a local computer, creating a local replica of these files for local or offline use. Rather than replicating all files, these services also offer the ability to access files remotely, which does not use up any storage space on the computer's local disk (other than to maintain a small on-disk cache to improve performance).

In addition to offering storage offerings for personal needs, cloud platforms also offer storage for enterprise data storage needs. There are a number of storage-only cloud providers that specialize in cloud storage services. Prominent providers include Dropbox, Box, and Backblaze. In addition, cloud providers such Amazon, Azure and Google also offer a full range of cloud storage services. Similar to compute resources, storage offerings can be viewed in terms of the delivery model.

- *Infrastructure Cloud Storage.* Infrastructure cloud storage represents the lowest-level building block of cloud storage, where storage is offered in terms of networked volumes. Such a storage volume can be attached to a cloud server, turning it into a networked-attached storage volume with a resident file system. It is also possible to directly read or write to cloud storage over the Internet via networked APIs. Infrastructure cloud storage comes several flavors.

- *Block Storage or Server disks:* In this case, cloud storage is exposed as block-level storage volume. It is primarily intended to be attached to a cloud server and provide a network-attached storage volume for the server. Storing data on network-attached storage as opposed to the server's local disk decouples storage from servers. The storage volume persists independently of the server and can continue to store user data even after the server is relinquished. The volume can be reattached to a new server in order to access data stored on it. A file system is typically created on the block storage volume and data is read or written using the file system interface.

- *Object-based storage:* Object-based storage provide the ability to read and write objects (i.e., blobs of data) to the object storage volume over a network. A typical interface

consists of get() and put() calls to access and write data. Object-based storage can be created and accessed independently of cloud servers, unlike block storage which can only be accessed by attaching the disk volume to a server.

- *File storage:* Cloud providers also offer a file-based storage in the form of a network file system. In this case, the cloud provider exposes the file system volume through a network file server that is operated by the cloud provider, allowing data to be accessed via network file system protocols such as NFS or SMB. Thus, block storage exposes a disk abstraction to a cloud server, while cloud file storage exposes a networked file system abstraction. This is analogous to a difference between a network-attached storage and a file system volume on a network file server.

- *Archival storage:* Cloud providers also support less expensive storage for archival storage and data backup needs. This is a lower cost storage tier for infrequently accessed long-lived data.

- *Platform cloud storage.* Cloud providers also support storage platforms for specified needs. We have already discussed storage platforms for photos, media and document storage, which are all examples of cloud storage platforms. In this case, the platform exposes higher-level interfaces for accessing stored data. For example, data can be viewed or modified via web browsers through a web-based interface. In addition, many platform offer two-way cloud synchronization which offers the ability to synchronize local files and folders on a computer with the cloud storage platform. Dropbox is an example of a storage platform that provides such data synchronization ability.

- *Storage-based Software-as-a-Service:* A final type of cloud storage is one where cloud storage is bundled with a higher-level software-as-a-service offering. Cloud-based data backup is an example of this type of cloud service, a backup service running on cloud servers also provided cloud storage as part of its service. Client computers use a backup client to backup their data to the cloud service and maintain data backups in the cloud.

Pricing models for cloud storage differ from those for compute resources. Cloud storage is priced by the amount of storage used, offering pay as you go pricing where volumes can be created or terminated on-demand. Unlike minute-level or hourly pricing for compute resources, storage use is charged by the month. In addition, certain types of storage such as object-based storage is also charged based on the number of I/O operations (gets and puts) as well as the Internet bandwidth used for I/O access. The cost of a gigabyte of storage also depends on whether it is block-based storage desired for storing online (hot) data or archival storage designed for storing infrequently accessed (cold) data. The latter type of storage is cheaper than the former.

Cloud Computing Applications and Workloads.

Cloud computing platforms are well-suited for several types of applications, and the set of cloud-based applications has grown steadily over the tears as new features have been incorporated into cloud platforms. Common cloud applications and workloads include the following.

- *Online web applications.* Since cloud platforms evolved from legacy web hosting platforms, web applications continue to be a common cloud workload. Web applications and services hosted in the cloud range from simple web sites to multi-tier web applications for online shopping, news, and entertainment, and from video streaming services (e.g., Netflix) to variety of web services (e.g., maps, email). Online web applications are interactive in nature and need to provide good response times to the end-user in order to provide a good user experience.

- *Batch applications.* Cloud platforms also run a variety of batch applications. Batch jobs are often periodic and can repeat at a set schedule (e.g., hourly or daily). For example, ACME online web store may run a set of nightly batch jobs to process orders, reconcile billing transactions, and update product catalogs. The use of cloud servers for software development is also common. In this case, servers provide a test and build environment to software engineers, with nightly builds of a software or periodic runs of regression test suites. There are many types of batch applications that are suitable for cloud platforms, and some even have dependencies where the completion of one job trigger one or more other tasks (e.g., a software build using make is a simple example of such dependencies). Unlike web workloads, key characteristics of all batch applications are that they non-interactive in nature and require good throughput or completion times.

- *Distributed data processing and analytic workloads.* Cloud platforms are increasingly used for running data-intensive ("big data") processing tasks where cloud servers are used to process large volumes of data, often in a distributed manner. One example of distributed data processing, which we will examine in detail in Chapter X, is Map Reduce-style data processing using platforms such as Hadoop and Spark. These workloads resemble batch-oriented data processing due to the need to optimize throughput. Such workloads also involve making analytic queries that involve processing a large amount of data to answer the query. Interactive analytic querying (e.g., drill-down queries) require query processing to be performed with latency constraints (to provide an result to the user in a timely fashion) and have resemble batch and interactive workloads. These workloads have been referred to as batch-interactive workloads. Spark supports such analytic queries in addition to map reduce-style batch data processing. Another example of distributed data processing and analytic workload comes from the use of cloud resources to establish a data warehouse or a data lake that stores an archive of an enterprise's business data. Data warehouses also run analytic processing tasks in the form of periodic batch or ad-hoc interactive queries over a large volume of data stored in the warehouse.

- *AI Workloads.* A new class of cloud workload involves running AI applications on cloud servers. AI workloads fall into two categories: training and inference. Training of machine learning models involves processing a large amount of training data to learn a model. Once trained, the model may be updated periodically as new training data becomes available. These workloads resemble batch-oriented distributed data processing and periodic batch tasks. A trained machine learning model is then deployed on a cloud server to perform inference, which involves making predictions over new data. Inference workloads are interactive in nature and have latency constraints. For example, voice assistants in smart speakers (e.g., Siri, Alexa) need to recognize voice commands and perform the requested command in real-time.

- *Scientific and High-performance Computing Workloads.* Scientific and high-performance

computing workloads arise in many domains such as physics, astronomy, medicine, and engineering. These workloads tend to be highly compute-intensive or data-intensive, or both. Until recently, such workloads ran on special-purpose clusters or supercomputers equipped with specialized networking and special-purpose processors (e.g., GPUs). However, as cloud servers have gained hardware such as GPUs and even FPGAs, such workloads have begun to use cloud resources, with the attendant benefits such as the ability to scale resources up or down based on workload demand. Cloud platforms are increasingly offering servers that are memory-optimized, I/O-optimized or equipped with accelerators such as GPUs to meet the needs to such workloads.