

Cloud Technologies: Executive Briefing

by Dan Appleman

Start Course

Bookmark

Add to Channel

Download Course

Table of contents

Description

Transcript

Exercise files

Discussion

Related

Cloud Services: Definitions and Motivations

What Is the Cloud?

The term cloud is used to describe a wide variety of technologies and approaches. What they have in common is a decoupling of applications and services from the computer resources that provide them. Let's start by considering what it actually takes for an organization to create or use software without the cloud using their own on-premises computer system. Every organization creates or uses software applications and services. These may be used internally, or applications and services provided to partners, customers, or the general public. In many cases, these applications run on computer servers purchased and maintained by the organization itself. That means that the organization is responsible for every aspect of the technology stack used to provide the applications. We'll call this on-premises when it comes to physical hardware, or in-house for software indicating that the responsibility for the software lies entirely in-house within the organization. Modern applications and services depend on numerous software libraries and frameworks. These all have to be kept up to date to ensure that the latest bug fixes and security patches are in place. They may make use of third-party applications and services such as database systems, web servers, and outside services. These also need to be maintained and

secured. The applications and their dependent software all run on top of an operating system, most often Windows or Linux. Any given application is ultimately built on a stack of software dependencies going down to the underlying operating system. This is fittingly called the software stack. The software stack ultimately needs hardware to run. Computer servers, of course, but typically there're also networking infrastructure, routers, switches, firewalls, file servers, and so on. And those computers need physical space and power, even air conditioning. Physical security is also critical, so Access control needs to be considered. Hardware fails, so backups are important. This includes backups of data, of the software, of application and hardware configuration, and even spare hardware, plus a process to verify and test those backups, and plans for recovery, and disaster recovery in the event of a major disaster. Both software and hardware need to be monitored. Is the application running? Is a cyberattack in progress? Have hackers accessed to the system? Are any of the hard drives showing signs of failure? Is the network up? Many things can go wrong, and it takes both technology and staff to monitor and maintain the entire technology stack. That's the job of the IT, or information technology, department. Scaling the system can be a huge problem. Computer resources need to be sized for the maximum expected load. Otherwise, at times of peak usage, the systems may become less responsive or fail completely, which can severely impact employee efficiency, customer satisfaction, and the organization's reputation, or lead to lost business or escalating costs. In short, having inadequate computing resources can be extremely costly. But it can also be costly to have adequate computer resources. In particular if the actual demand varies by hour, day, or month, it's very possible for most of the computer resources to spend most of their time sitting idle, which is very inefficient. Scaling on-premises systems, adding capacity, can also be time-consuming and costly. It can easily take days to bring up and configure a new server, longer if the existing physical infrastructure is not adequate, much longer if you have a complex purchasing process to go through. So adjusting the amount of computer resources to match demand is simply not practical for on-premises solutions. Remember, at the end of the day, all an organization really cares about is their own applications and services. It's almost shocking how much extra technology and cost--software, hardware, and staffing--is required to provide those applications. And anything that can be done to reduce those costs and associated risks is going to be compelling indeed. Cloud technologies in all of their forms decouple applications and services from the computer resources needed to provide them. Cloud technologies make it possible to outsource some or all of the technologies stacked to third parties using the internet as the communication channel that ties them together. They make it possible to connect or disconnect applications and services from computer resources enabling rapid scaling and de-scaling so that resources can be allocated to meet demand, then released when no longer needed.

Virtualization

In the early days of computing, the entire technology stack for an application was tied to one specific computer. The application relied on software libraries, frameworks, and possibly related applications. These in turn relied on infrastructure services and applications, for example, database systems such as SQL Server or MySQL and web server applications such as Internet Information Server or Apache. These applications in turn ran on operating systems such as Linux or Windows, which run on physical hardware. Each layer on the technology stack relies on the previous layer for services and functionality. The layers communicate through interfaces, clearly defined sets of protocols and mechanisms that allow the components in the system to work together. The beauty of this approach is that a particular component does not need to know the details of what is on the other side of the interface. Thus, an operating system can run on many different types of hardware, even on different processors, as long as the hardware has the necessary software driver to interface with the operating system. In other words, a core operating system does not communicate with the hardware directly. It relies on other software, drivers, to communicate with the hardware. But what if a driver instead of communicating with the hardware communicated with other software? Today's computers are powerful enough to stimulate hardware. And today's processors are designed specifically for this purpose. If a processor simulates hardware and provides a driver to that simulated hardware, then an operating system can run on simulated hardware and not even realize it. Using this approach, a computer could create multiple machines, each running its own copy of an operating system, each behaving as if it was a completely different computer. This concept, a computer simulating other machines, is called virtualization. The computer, that is the physical computer, is called the host machine. It runs virtualization software, also called hypervisor, such as VMware or Hyper-V, that manage one or more virtual machines. Each virtual machine is a simulated machine that runs its own copy of an operating system called a guest operating system. It's not unusual for a host machine to be able to run dozens of virtual machines. Virtualization does not just apply to general-purpose computers. Computer devices and appliances are really just specialized computers that rely on software to work. Internally, they too have a software stack and use drivers to connect to the hardware. Thus, they too can be virtualized. This means that it is possible to create virtual firewalls, virtual routers, virtual switches, and virtual networks. In fact, it is possible to take even the most complex network of computers and devices and simulate them on standard hardware that you might find in any data center creating an entirely virtual computer infrastructure. Or put another way, virtual computer resources need not be tied to a particular machine. They are defined entirely in software and are decoupled from the hardware. If a host

computer fails, that software can be instantly copied to another virtual machine. The virtual technology stack does not care what machine it is running on. Virtualization has fundamentally changed the economics and mechanics of providing and consuming computer resources. That, along with the general concept of technology layers and components that communicate through interfaces, and the availability of high-speed data communication through the internet, form the basis of what we call the cloud.

Cloud Technologies

Infrastructure as a Service (IaaS)

Virtualization makes it possible to simulate computers and other hardware devices and appliances. These virtual machines run on host computers. While there are applications for virtualization within an organization, the real benefits of this technology became apparent in 2006 when Amazon launched its EC2 or Elastic Compute Cloud service. This was the first mainstream Infrastructure as a Service product. That is, a service that allows organizations to rent virtual machines that run in the vendor's datacenters. While the technology itself is interesting, the success of Amazon and subsequent providers owes to the compelling economics of this approach. Infrastructure as a Service, or IaaS, allows organizations to outsource the hardware layer of their technology stack. What does this mean? First, it eliminates capital expense. IaaS is ultimately a pay-as-you-go approach. The challenge of scaling is largely solved. Organizations no longer need to purchase sufficient computing resources to handle peak loads. Instead, they can adjust the amount of computer resources they are buying based on demand. Bringing up new machines no longer requires purchasing and installing and configuring hardware, along with the necessary physical infrastructure, power, air conditioning, and security. With IaaS, bringing up a new machine involves just a few clicks of a button. A new machine can be ready with an operating system installed and ready to run in a matter of minutes. The physical infrastructure and all of its costs and headaches are handled by the IaaS provider. Downscaling is equally easy. Once peak load has passed, virtual servers can be downsized or discarded with no loss or waste. Even better, scaling can be automated. IaaS infrastructure can be controlled through software. Software can monitor demand and automatically spin up or tear down virtual infrastructure. IaaS simplifies the problem of backups and data recovery. Backing up hardware is difficult. It is, after all, hardware. And repairing hardware requires technicians and spare parts. IaaS largely eliminates

the problem of hardware failure. If host hardware fails, the IaaS provider can simply move the virtual infrastructure to a new host. Backing up virtual hardware is as simple as backing up software. This impacts the challenge of disaster recovery. If an organization's computer resources are entirely virtual, recovering from a major disaster that impacts one datacenter can be relatively easy, a matter of copying or activating the same virtual resources at a different datacenter. With appropriate preparation, an organization can recover from such an outage so quickly that nobody would even notice. Remember, IaaS is not just about computers. IaaS providers typically offer all kinds of virtual infrastructure-- firewalls, routers, databases, storage, and so on. Everything is available for a price. The economics of Infrastructure as a Service work primarily because of increased efficiency. Organizations no longer have to invest in sufficient computer hardware to handle peak loads only to have those resources sitting idle for much of the time. They no longer have to absorb the costs of lost business and employee inefficiency caused by not having sufficient computer resources on hand. Instead, they pay only for what they need as they need it. At the same time, cloud vendors benefit from the economics of scale, supporting vast numbers of identical servers in datacenters, each of which can be heavily utilized by optimizing the deployment of virtual machines on those servers. The costs of physical infrastructure and security can be amortized across many machines and many customers. This increase in efficiency allows the cloud vendor to make a healthy profit while their customers reduce costs. This economic logic explains why organizations whose hardware is fully utilized may not see cost savings by switching to IaaS. Larger enterprises who have a high baseline of computer use often do maintain their own data centers. In some cases, they use virtualization internally, creating a private cloud where the organization's IT department provides virtual computer resources to various divisions and departments within the organization. Some organizations combine the two approaches creating a hybrid cloud in which a baseline of computer resources is provided internally, but additional resources are built out on a third-party service when they are needed or to scale based on varying loads. These internal and external resources can be tied together with a virtual private network allowing an organization to access on-premises and cloud-based resources seamlessly. IaaS can substantially reduce the cost of computer resources but, surprisingly, has less of an impact on staffing requirements. While there may no longer be a need for IT staff to repair hardware and string cables, IT staff is still required to manage virtual infrastructure. Some traditional tasks remain. An IaaS vendor is responsible for the patching and security of the host operating system, but they do not generally maintain the guest operating system. All of those virtual machines and their operating systems must be kept up-to-date and secure. In addition, the skills required to manage the IAS infrastructure itself can be quite demanding. The ability to create and destroy infrastructure within minutes means that untrained or inexperienced

individuals can incur significant costs or cause serious problems very quickly. So staffing costs may actually increase due to the need to recruit and train individuals qualified to work with cloud infrastructure. Ultimately, IaaS involves outsourcing the hardware level of the technology stack to third parties. Given the demonstrated benefits of doing so, one can't help but ask, Is it possible to outsource other layers of the technology stack as well? The answer to that brings us to the next cloud technology--Platform as a Service.

Platform as a Service (PaaS)

Where Infrastructure as a Service provides a mechanism to outsource the hardware part of a technology stack, the idea behind Platform as a Service, or PaaS, is to outsource some or all of the software and hardware components required to run the organization's own applications and services. In theory, an organization can define all of the components that their applications need--databases, web servers, storage, queues, authentication services, and so on--and have those provided by a cloud service provider. The organization can then just deploy their own applications to the cloud, and they will work. A cloud service provider is responsible for maintaining all of those underlying services, along with the operating system and hardware, handling security, backups, automatic scaling based on demand, even disaster recovery. The organization pays only for the resources they use, primarily compute time, storage, and bandwidth. That's the theory. In practice, there are many different approaches and technologies that cloud vendors use to provide Platform as a Service capabilities. They vary in cost, capabilities, features, complexity, and the amount of control an organization has over their operation. Let's look at several of the approaches. One PaaS approach is to add automation to the deployment of software and the creation and management of groups of virtual machines. The organization defines the resources needed for the application, and requests the service provider to provide that environment, essentially creating a software platform. The organization can then use configuration and automation to control the behavior of the platform and to implement desired features like backups. This is the approach taken by Amazon with their Elastic Beanstalk product. This approach offers a great deal of flexibility at the cost of additional complexity. Another PaaS approach that provides a higher level of abstraction eliminates the need to worry about virtual machines at all. The organization defines the resources needed for their application and services and deploys their software. The cloud service spins up virtual machines as needed, loads and installs the required software services, and the application is available in minutes. An organization can define the amount of computer and storage resources they want to make available to the application, or configure auto-scaling to adjust those within a range as needed.

Azure's App Services and websites are a good example of this approach, as is Google's App Engine and Heroku, which actually runs on top of Amazon Web Services. A third more recent PaaS approach is ideal for organizations who need or build applications consisting of small independent operations and services called micro services. These small applications are designed to run in a specific software stack and can be deployed to a cloud service that will simply run the code on request. The organization need only concern itself with their own code. The cloud provider handles everything else. A part of most Platform as a Service solutions involves an application defining the resources and services that it needs in order to execute. This can take the form of a script, a list, or a template that actually contains some or all of those resources. The cloud service can read the requirements or components from the template and create a running instance of that application in what is called a container. Just as virtualization software allows hardware to be shared by multiple virtual machines, containerization software allows a single operating system and related services to be shared by multiple containers that are created based on resource templates or images. This can be quite a bit more efficient than virtual machines where each virtual machine has its own copy of the entire operating system. Because containers define everything an application needs to run, they are ideal for Platform as a Service applications. Containers can be quickly deployed to one or multiple servers making scaling easy to implement. They are often used under the hood, even if an organization is not making explicit use of container software. They also lead to portability. If an organization has built a container for their application using Docker or other container technology, they can quickly run it on everything from a development desktop to an in-house server to a PaaS cloud service or a virtual machine with almost any cloud vendor. Containerization is, thus, a core technology for building hybrid PaaS solutions allowing organizations to build one set of software that can execute on their on-premises service and be easily deployed to PaaS services on the cloud when needed for scaling or as a disaster recovery strategy. Unlike Infrastructure as a Service, with Platform as a Service, it is possible to see significant savings in terms of IT staffing. While managing PaaS offerings does require skilled personnel, it is typically easier than managing IaaS resources, potentially much easier, depending on which approach you use. With most PaaS offerings, there is no need to maintain virtual machines at all. The responsibility for patching, configuring, and securing operating systems is entirely with the cloud vendor.

Software as a Service (SaaS)

If Infrastructure as a Service allows organizations to outsource hardware resources, and Platform as a Service allows organizations to outsource the technology stack on which their applications

and services run, the obvious next step is to outsource everything, including the applications and services that the organization itself uses. This approach is called Software as a Service, or SaaS. Almost every individual and organization today makes use of Software as a Service products--Gmail and Google Docs, online conferencing such as GoToMeeting, Zoom, and BlueJeans, online surveys such as SurveyMonkey, Quicken Online for accounting, the list goes on. And new services appear almost every day. In a traditional SaaS application, all of the application software runs on the vendor's servers. Users interact with the software through a web browser and occasionally a browser plug-in. Some vendors stretch the term to include subscription services that combine some cloud-based software with a subscription to software that is actually installed locally. Adobe Creative Suite and Microsoft Office 365 are examples of this. Some vendors offer their software either as SaaS or for on-premises installation. Jira by Atlassian is an example. SaaS solutions can have multiple benefits. Given the wide array of available solutions, it's quite possible for an organization to not have any servers at all. No servers means no staff to maintain the servers and no worries about scaling, backups, disaster recovery, or physical security. These are all the responsibility of the vendor. Supporting users can be dramatically simplified as most of the work will be done in a web browser. Direct IT costs can be dramatically lowered than other approaches, though obviously some of those costs are not truly eliminated, just shifted into the SaaS application's subscription fees. However, as with any choice, this one comes with trade-offs. SaaS solutions tend to be unique and proprietary. Choosing one represents a commitment, so it is essential to investigate whether the vendor is trustworthy and stable as migrating to another solution can be quite costly. Just because a software solution is on the cloud does not mean it is simple. Today's SaaS applications, especially enterprise applications, can be incredibly complex and highly configurable and customizable. Some allow organizations to deploy custom software to truly individualize their behavior. Examples of these are marketing and sales automation systems, such as Eloqua or Salesforce. Some allow so much customization that it is possible for an organization to create its own applications within the software that are distinct from the main SaaS application. Thus, Salesforce, in addition to being a SaaS application, is arguably a PaaS offering as well, or maybe Software/Platform as a Service because it's very different from a typical Platform as a Service offering. It is easy and common to underestimate the costs and staffing required to support SaaS applications. The promise of no hardware or software infrastructure implies that the IT staff has no role. But, in fact, these complex applications frequently require an army of administrators and even developers to customize and maintain the service to meet an organization's changing needs and to support users and to maintain operational security. The responsibilities for managing SaaS applications are not always clear. In some organizations, the IT department takes the lead. In others, the department that purchased

the SaaS solution is responsible. And in some companies, the responsibility is shared or fought over. So this is an issue that should be addressed in any SaaS deployment.

Evaluating Cloud Vendors

Cloud Vendors: The Big 3... 4... 5... 6?

When thinking about cloud vendors, most people think first of the big three-- Amazon Web Services, or AWS, Microsoft Azure, and Google Cloud Platform. So let's consider them first. All three vendors offer similar services, and any places where they don't are likely to change quickly as they are constantly adding new features to keep up with each other. All three have been adding cloud services and specific technology domains, such as artificial intelligence and Internet of Things. All three have multiple datacenters in multiple geographic locations. AWS has its roots in Infrastructure as a Service and, indeed, offers a vast array of infrastructure products beyond basic virtual machines. It is currently the largest player in this space and often the first choice of those who are truly looking to build out cloud infrastructure. Microsoft Azure started out with a Platform as a Service offering but like the others now offers IaaS as well. Still, PaaS is in their DNA, in particular services around Microsoft technologies. It is often the first choice for developers using the Microsoft .net framework and Visual Studio, and for companies using Microsoft's enterprise software. Google's roots lie in managing and searching vast amounts of data, so it's no surprise that their cloud services shine in that area with multiple database offerings that can support massive scale. All three support today's standard technologies from operating systems to containers to popular languages, so it is quite feasible to migrate between them or even combine them to form solutions, but doing so does incur the added cost of training staff on multiple platforms. There are, of course, a great many other cloud vendors, some which are also quite large and who offer different and even customizable levels of service allowing organizations to fine-tune which tasks they wish to outsource and which they wish to keep in-house. Many vendors focus on providing management services and technologies beyond or even on top of the cloud services provided by the big three. With managed IaaS services, vendors such as Rackspace take on the responsibility of configuring, securing, and updating individual virtual machines, freeing organizations to focus on their own applications and services while maintaining the flexibility of a virtual machine-based solution. These vendors may also offer consulting services to design, size, and implement cloud-based solutions. At the one end, almost every web hosting company offers

managed virtual private servers, in many cases with selectable management features often at astonishingly low prices. While AWS, Azure, and Google often come to mind first when it comes to cloud vendors, it would be a mistake to ignore another set of cloud vendors, those whose strengths lie in providing solutions at the enterprise level. Companies such as IBM, Oracle, SAP, and Salesforce provide many of the same cloud services as the big three but with their armies of engineers and consulting partners are positioned to offer business-specific advice and complete solutions as compared to just cloud technologies. It can also be an excellent choice if you are already using one of their Software as a Service offerings.

Evaluating Requirements and Costs

When choosing cloud vendors, there's often a temptation to compare them against each other especially in terms of features and available technologies. This is a mistake. A better choice is to evaluate cloud technologies against the requirements at hand. This means, of course, that it's essential to figure out those requirements. What is the problem that the organization is trying to address? What is the anticipated scale? What are the available resources both in terms of technology and in terms of available personnel? What other cloud services may be needed? Be sure to consider both immediate needs and future growth and improvements. In some cases, it will become quickly apparent that a particular cloud vendor is a close fit to the requirements making the choice relatively easy. If that is not the case, which is quite common, the decision can be made based on cost. A large part of the drive towards using cloud technologies is there potential to reduce costs, but this potential cuts both ways as they have the potential to increase costs as well. Cloud technologies are all based to various degrees on a pay-for-use-model. And organizations that use large amounts of cloud resources can get hit with surprise bills that are much higher than expected. While individual resources may be inexpensive, they can add up quickly. Vendors often charge for CPU time, data storage, data transfer, database lookups, monitoring, even DNS lookups. And while pricing tends to be transparent, estimating costs requires a good understanding of the resources consumed by applications and services. And this is not always practical. Cloud services can scale quickly based on demand, but that means costs can scale quickly as well. More than one company has been surprised by charges for resources that a staff member created but forgot to delete once their work was done. And don't forget IT costs. As mentioned often throughout this course, cloud technologies require administration and maintenance that can be different but in many cases just as complex as maintaining on-premises servers. When moving to the cloud, existing IT staff may need significant retraining, and it may be necessary to hire outside consultants with specific expertise during the process. If an organization

is not prepared to bring in the necessary staff for training, choosing an enterprise cloud provider that offers a high level of solution design and management can be a good alternative. Taking the time to define requirements and choosing a cloud platform or vendor that meets those requirements and investing in the personnel or services required to design, implement, and maintain those solutions will dramatically increase the odds of a successful transition to the cloud. Pluralsight has numerous courses on cloud technologies including not only AWS, Azure, and Google, but Oracle, SAP, and Salesforce as well. These include courses both for developers and for IT staff and can play a significant role in preparing teams for working in the cloud. With that, I'd like to thank you for watching this Cloud Technologies Executive Briefing, part of a growing series of executive briefings here on Pluralsight. We hope you enjoyed this course. If you're interested in more content for technical leaders, content we keep short and focused with the up-to-date information you need to be informed and make decisions but without getting buried in the details, find other courses like this at [plrsig. ht/exec](https://plrsig.ht/exec).

Course author



Dan Appleman

Dan Appleman is a well known author, software developer, and speaker. Currently the CTO of Full Circle Insights, he is the author of numerous books, ebooks, and online courses on various topics...

Course info

Level	Beginner
-------	----------

Rating	★★★★☆ (46)
--------	------------

My rating	★★★★★
-----------	-------

Duration	0h 31m
----------	--------

Released	31 Jul 2018
----------	-------------

Share course

