Azure vs. AWS

Battle of the Cloud Giants

Learn the Benefits and Drawbacks of each Provider.

comport

TECHNOLOGY SOLUTIONS

As of 2020 AWS is the largest cloud-computing platform in the world. However, Microsoft's Azure is in a solid second place and appears to be gaining on AWS. Although Microsoft doesn't release Azure sales figures to the public, careful evaluation of Microsoft's earnings reports allows an estimate of it's scale and growth. Synergy Research Group's comparison of AWS vs Azure shows that AWS had 33 percent of the combined IaaS and PaaS market in Q3 2019, compared to Azure's 16 percent for that time period. other in specific areas.

First a little bit of history, AWS launched in 2006, based upon Amazon's expertise at operating a massive ecommerce organization, and was already a well-established cloud platform by the time Azure launched in 2010. As a result, Microsoft had to enter this space sooner than previously planned to prevent its enterprise customers from moving their on-premises infrastructure to IaaS platforms hosted by AWS and other competitors. Enterprises are Microsoft's core customers for its Windows ecosystem, so they needed to deliver an IaaS solution quickly.

As a result, the Azure interface and automation methods are familiar to Microsoft Professionals, as it is tightly integrated through Powershell with both Microsoft Applications such as SQL and the Windows Server OS. Once Microsoft did decide to commit to cloud computing, it did so fully. This strategy allowed Azure to quickly break away from smaller competitors, despite its relatively late start.

Getting Started

Because of Microsoft's focus on the enterprise, Azure will probably be your best bet for traditional businesses where your IT staff are new to cloud computing but are already familiar with the Microsoft stack. On the other hand, due to their lineage in e-commerce and cloud native business, AWS may be a better choice for organizations that principally do business through the internet or are Dev centric and highly automated.

It's particularly important to consider your team's skills when it comes choosing the best interface. AWS is more technically oriented and has a Linux feel to it, although it does have a deeper functionality than Azure. Microsoft has greater experience in designing for end users, so its interfaces are simpler and more intuitive. If your team has existing experience in automating Windows environments using Powershell, those skills may be portable to your Azur environment. It's important to note however that both Azure and AWS offer serverless technologies.

The use of virtual machines (VMs) also illustrates the differences in design philosophy between AWS and Azure. For example, Azure merely requires you locate a node called "virtual machines" and deploy it. In comparison, deploying VMs in AWS requires multiple components like EC2, VPCs, ALBs or ELBs. Team members will thus have an easier time with AWS if they have experience with Linux.
It's important to note however that both Azure and AWS offer serverless technologies.

Azure
Getting started in Azure begins with creating group of related functions known as a Function App. You'll need to create each function within the Function App, which has access to a storage container. Azure functions are event-driven and provide computing resources on demand, allowing them to extend the capabilities of the Azure platform by executing code in response to triggering events. These events can come from other Azure functions, on-premises systems and SaaS products. Developers can write Azure functions in multiple languages, including C#, F#, Java, Node.js and Python.

AWS

AWS users get started by creating Lambda functions, which only requires them to specify the function's development environment without worrying about the allocation of resources. This capability helps users abstract logic into a cloud function that they can then scale, limited only by the third-party services that the function connects to. Common functions that users perform with Lambda include reading and writing data to an AWS component, creating a web application and data processing.

Beginner Deployment Scenarios

Running VMs in the cloud is one of the most common deployment scenarios for beginners. Azure has a slight edge on this issue because this is a common requirement for end users, which is what Microsoft focuses on. AWS has more components that aren't necessarily of direct interest to end users, making it more complex to deploy. For example, AWS only has virtual storage, which goes away when the VM terminates. In other words, you lose changes when you reboot your VM. This feature of AWS isn't intuitively obvious, and as such it has a steeper learning curve than Azure. In this example, the appropriate design is to utilize EBS Volumes that will allow you to keep the EBS storage after the VM Terminates – this design is different than traditional on-premise workloads, and can introduce complexity to administrators that are new to Cloud workloads.

Azure

VMs provide users with greater control in Azure, typically when it comes to migrating applications to the cloud. You can install anything the application needs to run on a VM, from antivirus software to the OS itself. However, this capability also means the users are responsible for ensuring the applications remain available, while meeting their performance and security requirements. This process often involves striking a balance between control and responsibility, which can be challenging for new users.

In addition, Azure users can roll out VMs with images which they can install on premises. However, migrating these VMs to the cloud is a comparatively slow process, since these images are often very large. Provisioning and starting the VM also requires time, which may be prohibitive. This approach is more complex but may be more flexible than creating a VM on the cloud from scratch.

AWS

AWS users should begin their cloud deployment with either the free tier, or one of the simpler services like S3. Once they become comfortable with this process, they can progress to services with progressively steeper learning curves to discover how AWS works before making a final decision on which platform to use. As is the nature of IT, your projects are often due yesterday, making it critical to ramp up on AWS first before committing to it.

A series of small test sets containing organizational data is often the next step after developers have gained basic experience with AWS. They may use these sets to perform simple tasks that help determine the feasibility of using AWS and its performance. Administrators typically find that working in AWS is a different experience, making it important to avoid moving too quickly.

Evolution of Products

AWS is more mature than Azure, so it rarely has major architectural changes. However, there are new products and services are being released from AWS on a regular basis. Azure, on the other hand, is still evolving at a fairly rapid pace, which may require users to change their strategies after they've already implemented Azure.

Azure
The release of Azure VMware Solution (AVS) in May 2020 was one of the biggest changes to Azure for that year. This solution is a first-party service available in Azure on bare metal servers. It allows users to seamlessly expand or migrate their on-premises VMware workloads to Azure. This capability eliminates the need to change the architecture or operations of applications, greatly reducing the cost, effort and risk of migrating to the cloud.

AWS
The biggest changes to AWS in 2020 include the release of Amazon AppFlow in April. This fully managed integration service securely transfers data between AWS services and third-party SaaS services. Amazon also added Amazon Honeycode to AWS in June, which is a no-code platform for developing web and mobile applications. AWS also offers first-party services. For example, VMWare Cloud (VMC on AWS) runs on AWS bare metal servers and provides a native dedicated VMware experience, hosted on demand within AWS datacenters.

DevOps

Both Azure and AWS can create an agile development with API’s that can be used for DevOps. The AWS Elastic Beanstalk orchestration service is a great choice for implementing DevOps if you’re already comfortable with the Linux environment. Azure has a more flexible approach since it isn’t limited to Linux. However, DevOps is still more difficult in Azure due to the more mature tools available within AWS.

Azure
Microsoft provides several features and services that can help your team implement a DevOps environment in the cloud, including Azure App Services, Azure DevOps Projects and Visual Studio Team Services.
Azure App Services are a collection of Azure services that host APIs and web applications. Developers can automatically deploy their applications to Azure App Services whenever they need to push up a change. This capability requires the source code to be under some type of source control.
Azure DevOps Projects is currently available as a preview. It guides the user through the Azure Portal, making it easy to configure continuous integration, and continuous deployment (CI/CD). Azure DevOps Projects is more of a UI than a service that sets up a build-and-release pipeline in Visual Studio Team Services (VSTS). This approach means that builds and releases occur within VSTS rather than Azure DevOps Services.

VSTS is the most powerful and flexible way of creating CI/CD pipelines for Azure. It's essentially the online version of Team Foundation Server and is a highly mature product. You can use VSTS to host source code and deploy applications to Azure, whether it's installed on-premises or on cloud platforms. You can also use it to track assignments through their workflow, allowing you to associate them with deployments and conduct performance tests. These capabilities only require access to Visual Studio, which includes templates to help you get started.

AWS

AWS provides a collection of services that allows users to implement DevOps practices quickly and reliably. These services perform tasks such as provisioning and managing infrastructure, monitoring application performance, deploying application code and automating software releases. These capabilities shorten the software development cycle, allowing organizations to better serve their customers.

All of these services are available with an AWS account, and there's no software to install or other setup required. They also allow you to use AWS resources more easily, allowing you to focus on developing software rather than worrying about infrastructure. AWS services are highly scalable since users can manage a single instance just as easily as thousands. The ability to provision, configure and scale resources helps users maximize the use of those resources, which is a core value of DevOps.

AWS provides multiple methods for using its services, including APIs, SDKs and the AWS Command Line Interface (CLI). You can also use declarative AWS CloudFormation templates to provision resources and model your infrastructure. In addition, AWS helps you build your infrastructure more efficiently with services that deploy applications, develop workflows, manage containers and configurations. Another way that AWS supports DevOps is with its Identity and Access Management (IAM) feature, which provides a highly granular method for control over who can access resources and how they must access them.

AWS also supports solutions from many partners that integrate with and extend AWS services. Developers routinely use their favorite third-party tools to build complete solutions. AWS also supports many open-source tools.

Hybrid Cloud

Most organizations today are looking for some type of hybrid cloud environment, making this issue an important one to consider when choosing a cloud platform. In particular, you should determine the platform that will work best for your existing infrastructure.

Microsoft has pretty good deal of effort into ensuring Azure has strong hybrid conductivity, due to its long experience with on-premises environments. Many of its enterprise clients require a tight integration between their on-premises and cloud infrastructure, especially those in highly regulated industries like the financial sector. Clients in these industries will probably prefer Azure for this reason.

However, Azure and AWS both have tools for linking on-premises infrastructure with virtual networks, including dedicated hybrid tools and VPN software. One notable difference between these two platforms is the way they handle identity gateways. Azure favors the use of Active Directory, while AWS offers support for open-source alternatives or Active Directory.

Azure
Azure has always been hybrid by design, allowing it to provide enterprise customers with the flexibility and consistent experience they need to meet their business needs. Many of the world's leading businesses already run on Azure, which provides a seamless approach to managing applications from any location. This capability is particularly important for clients that store applications in multiple places, including on-premises, in the cloud and on edge devices.

Microsoft has recently made some major additions to Azure's hybrid capabilities. It launched Azure Arc in November 2019, which provides users with greater flexibility in developing their platform. This service allows other Azure services to run from any location and has been adopted by customers in many industries. Major Azure Arc customers currently include the following:

Africa’s Talking
Avanade
DexMach
Ferguson
Fujitsu
KPMG
Siemens Healthineers
These and other Azure users manage their resources more efficiently with Azure Arc, especially in distributed environments. They also use Azure Arc to access Azure data services from their on-premises infrastructure.

AWS
Enterprises often need a hybrid cloud architecture to integrate a mixture of on-premises and cloud operations across a broad spectrum of use cases. VMware Cloud is an integrated cloud solution developed jointly by AWS and VMware. This scalable, secure service allows organizations to migrate and extend their on-premises VMware vSphere environments to the cloud by running them on Amazon Elastic Compute Cloud (EC2) bare metal infrastructure. Workloads may then be migrated from your on-premise VMware environment to the VMware Cloud on AWS through the use of VMware HCX.

AWS Outposts is another hybrid cloud solution that provides AWS users with the same environment they already know, whether the new infrastructure is an off-site data center, on-premises facility or co-location space. This solution migrates the entire environment, including the infrastructure itself as well as its APIs, management tools and operating model. AWS Outposts also allows users to run many AWS services, including Amazon EBS, Amazon EC2, Amazon EKS and Amazon S3.

IaaS

Many organizations move to the cloud specifically for the purpose of obtaining IaaS. A comparison of Microsoft Azure vs AWS shows that both solutions offer this service, although their approaches are quite different.

Azure
Microsoft had to match AWS’s IaaS capabilities if it was going to compete in the cloud-computing space. It also wanted to provide Azure with capabilities that would allow it to address areas that were underserved by AWS. Because of AWS early entry into cloud services, AWS focuses on the market segment that has already embraced the idea of placing infrastructure on a public cloud. Azure’s strategy has been to target users who have not yet migrated their operations to the cloud, or who have an immature cloud strategy.

Microsoft’s general strategy has been to provide a pure IaaS infrastructure in addition to integrating on-premises and public cloud infrastructures. This strategy accepts that not every client will want to move its entire infrastructure to a public cloud. For example, organizations in the government, finance and health sectors need to maintain physical control of their infrastructure for regulatory compliance purposes.

Azure Stack is a complete Azure solution that includes everything users need to operate an IaaS model either as an on-premises cloud or tightly integrated hybrid cloud. It includes both the Hyper-V hypervisor and the additional technology needed to bring on-premises and cloud resources together in a seamless whole. Microsoft requires control over the user's environment to make stable, dependable updates to Azure, including both hardware and software. That means that only certain vendors can deliver Azure Stack, which currently includes Dell, HP Enterprise, and Lenovo.

These resources generally consist of compute and storage resources. Microsoft also provides their clients with Operating Systems, so it has complete control over this aspect of the client's infrastructure. It is therefore able to provide the real-time migration of VMs between the client's premises and the cloud out of the box. In comparison, AWS requires an additional layer of abstraction to accomplish this task.

Azure provides the seamless integration of data storage between data centers, which is particularly useful for requirements like data recovery and regulatory compliance. AWS also offers these capabilities, but they aren't integrated into the OS as tightly as Azure. Big data is one aspect of storage where Azure falls short of AWS, since Azure requires premium storage to achieve this capability.

The technical capabilities of Azure and Azure Stack are very similar, but their pricing models are quite different. Azure is strictly a pay-as-you-go service, meaning that its cost is entirely an operational expenditure (OPEX). Azure Stack incurs an ongoing cost as well, but it also requires a sizeable capital expenditure (CAPEX) to establish the Azure Stack environment. The business case for acquiring Azure Stack will therefore be more difficult to make than the one for Azure.

AWS

Amazon EC2 offers a large number of instance types, which are configurations for its virtual servers. This capability is most beneficial for workloads that are resource intensive since each instance type is optimized for a particular use case. In addition, instance types are also available in several sizes. Organizations using Amazon EC2 are thus able to tune their environment for almost any workload, regardless of size or resource requirements.

The total server capacity of AWS is about six times that of all its competitors combined, including Azure. This huge capacity helps drive down the cost of IaaS, justifying greater usage by clients. As a result, most of the innovations in IaaS appear in AWS first, before propagating to Azure and other platforms.

The difference in size may make AWS the most cost-effective IaaS option, depending on the workload. Unfortunately, a direct comparison of cost between AWS and Azure is very difficult due to the significant differences in their pricing models. However, additional research may allow you to determine which pricing model is most appropriate for your workload. Spot Instance Pricing is one feature of AWS that makes it more favorable to compute-intensive workload. This feature allows Amazon EC2 clients to bid on unused capacity, which can result in a savings of up to 90 percent.

AWS also offers multiple storage options like Azure. Amazon EBS is much faster for big data, so users who need to move large amounts of data will find AWS to be the more attractive option. This is especially true for data sets that are so large that transfers over the internet are no longer cost effective.

In these cases, AWS offers options to physically transfer data between an on-premises server and the cloud. These include AWS Snowball Edge, which is a device that can transfer up to 100 Tb of data. AWS Snowmobile can transfer up to 100 Pb of data. This device is enclosed in a 45-foot-long ruggedized shipping container that's transported on a semi-trailer truck.

PaaS

Azure and AWS generally offer similar PaaS capabilities. One of the most obvious differences is that AWS is more likely to provide newer APIs, since it has a larger number of vendors. In comparison, Microsoft is still developing its partner network for Azure, although it does have closer ties with applications outside its own ecosystem than AWS. As a result, Azure could be a better choice for users who need access to those solutions.

Azure

Azure provides PaaS services under a collection services known as Azure App Service. It provides hosting services and tools that allow users to streamline their application development and deployment process, including development frameworks and programming languages. App Service consists of the following tools:

Azure API Apps
Azure Web Apps
Azure Web Apps for Containers

These tools are particularly useful for organizations that need a high degree of flexibility in customizing their PaaS environment. In fact, Azure comes closer than any of the other major cloud platforms to providing a turnkey PaaS solution in one package. Azure is therefore the best choice for organizations needing an easy way to implement a PaaS.

AWS

AWS provides less integration between its PaaS services than any other major cloud vendor. Users who want PaaS capability on AWS must join several services together to build a complete pipeline for developing and deploying applications. This requirement means that obtaining PaaS on AWS will have the greatest appeal for organizations who want to pick and choose their own tools.

They must also be willing to integrate multiple AWS services into their infrastructure, as no single Amazon cloud service qualifies as PaaS. However, clients can integrate the following services to create a complete PaaS:

AWS Cloud9
AWS CodePipeline
AWS CodeDeploy
AWS Elastic Beanstalk
AWS Cloud9 is an integrated environment for developing applications on the cloud. Developers use
AWS CodePipeline to build and deploy their applications. This service supports a range of hosting options, including VMs in EC2 and containers in Amazon Elastic Container Service (ECS). AWS CodeDeploy allows enterprises to automate their software deployments to AWS compute services. Developers can use AWS Elastic Beanstalk to upload code for web apps and services, without worrying about the details of provisioning, deployment and load balancing.

Team Skill Evaluation

A team that's already familiar with Microsoft products will have an easier time in transitioning to a cloud environment in Azure. Similarly, a team used to working in Linux and OSS package will get up to speed faster with AWS. Regardless of the specific platform you choose, the key to continuing to developing technical skills in the cloud is to understand the difference between on-premises infrastructure and cloud architecture.

Azure

Microsoft has a long history in providing technical education and its approaches have continued to evolve with technological advancements. Early Microsoft training consisted of delivering shrink-wrapped CDs to its clients every few years, but Azure cloud services are now updated daily. Microsoft's training offering were initially quite narrow and only focused on using specific products, with infrequent changes to content. Today, its Azure training tends to emphasize the performance of key roles and the skills they require. Furthermore, Microsoft typically updates its Azure training every couple of months to reflect the rapid changes in the cloud-computing landscape.

Microsoft obtains its training material for Azure by analyzing data on key roles within modern enterprises of all sizes, typically Microsoft's internal and external partners. This process is known as a Job Task Analysis, or JTA, which provides a list of required skill for each role. Microsoft performs JTAs on an ongoing basis to ensure the capabilities needed to perform each role effectively remain current as technology evolves.

AWS

AWS has a steeper learning curve than Azure, especially when it comes to setting up the initial services and hosting a website with a complete set of APIs. Users who want to implement capabilities beyond the bare essentials will also need to setup additional services. For example, users needing CI/CD may need to install third-party solutions like Jenkins or Travis CI.

Web developers are generally finished with AWS once they configure their environment, but application developers will also need to know how to use their code repository to push new code to the cloud and merge it with the current code base. At this point, AWS has a minimal impact on developers, as the automated CI/CD process will then take over and test the code before deploying it on AWS. Developers can then focus on their primary task of writing and modifying code.

Pricing

Reducing the CAPEX of IT infrastructure is typically the driving force behind the decision to move to the cloud. The increasing competition between major cloud providers has been steadily pushing their prices downward for some years now. An apples-to-apples comparison of Azure vs AWS cost remains difficult due to the differences in their pricing models.

AWS has a more flexible pricing model than Azure, but the greater number of available options also means that AWS customers are more likely to pay more than they need to. Both solutions provide introductory tiers with restricted usage limits at no charge, allowing users to try their services before making a commitment. They also provide credits for new users, which is especially attractive to start-ups. Azure and AWS both have a pay-as-you-go model, but Azure charges by the minute while AWS charges by the hour.

While a reduction in costs may have been a driver to move workloads to the cloud, it shouldn't be the primary reason. It'll will take high degrees of IT governance and cost avoidance to truly reduce the cost to run within the cloud. It's designed to be extremely easy to use and consume and therefore will tend to make bills increase at the same rate.

AWS
AWS's rates generally drop as the customer's usage increases, although the specific benefits depend on the pricing model. Reserved Instances is a pricing model that involves paying a cost upfront based on use, which allows the user to obtain a fixed rate for the next one to three years. On-demand Instances allow users to pay only for what they use without any upfront cost, while Spot Instances let users bid on extra capacity as it becomes available.

Azure
The pricing options for Azure generally favor shorter commitments than AWS. For example, Azure users can choose between paying in advance or by the month.

Conclusion

It's essential to understand what you're getting in a cloud solution before committing to it. These factors can be complex, so it's often helpful to consult with an expert who can explain the differences between Azure and AWS. Fortunately, you don't necessarily to choose only one provider. Comport can help you decide the best location and vendor for your workloads, so give us a call today.