CloudCheckr

GUIDE

The Enterprise Buyer's Guide to Public Cloud Computing

When assessing enterprise compute options on Amazon and Azure, it pays dividends to research the number of different services at your disposal—because, with a clear understanding of the choices available, your teams will be better equipped to make a successful migration to the cloud.

As the two leading public cloud services, Amazon Web Services (AWS) and Microsoft Azure have a combined share of the global IaaS market of more than 40%. They offer a vast array of cloud products and features, with solutions to suit practically any type of enterprise IT requirement. So, when it comes to choosing the right platform for your cloud-based applications, it makes smart business sense to include both AWS and Azure in the shortlist of IaaS providers for your organization.

Both vendors offer equally compelling propositions and share many common capabilities. But, at the same time, they differ significantly in terms of their approach to service delivery, pricing structures, and the way in which their products solve organizational IT challenges.

Whether you're planning to build a multi-cloud or hybrid-cloud environment, or prefer to host applications exclusively on a single vendor platform, you'll need to consider a whole variety of aspects of the two leading services—from storage and content delivery to network and security.

However, in this paper, we will focus on cloud computing. We'll take a look at the each of two leading vendor's standard compute services, virtual machines, before moving onto other compute offerings and alternatives to the on-demand payment model.

AWS and Azure: A General Overview

Amazon and Azure have a widely different history, which is reflected in their cloud offerings.

Amazon was a product of the Internet revolution in the late 1990s. Turning the infrastructure and technology from its eCommerce operations into a commercial proposition led to the launch of Amazon cloud in the early 2000s. As a result, it has the look and feel of its online retail business and is more oriented towards the **self-service model** and born-in-the-cloud applications. It also boasts a large ecosystem of third-party vendors and strong support for open-source applications.

By contrast, Microsoft has had a long history of working with enterprises. So it takes a more traditional **hands-on approach** to service delivery—through its network of sales representatives. It has strong hybrid cloud capabilities and is seen as a more natural choice for enterprises with existing Microsoft investments.

A New Era of Cloud Maturity

It's now more than 10 years since the origination of the cloud, when AWS launched its flagship storage service **Amazon S3** in 2006, quickly followed by compute service **Amazon EC2** later that year.

The cloud is now entering a new era of maturity with a recent string of high-profile contracts and strategic alliances aimed at promoting wider enterprise adoption of the cloud.

One significant development came in 2015, when Microsoft penned a deal with VMware to support its virtualized workloads. Fast on its trail, AWS also signed its own contract with the Dell-owned company.

VMware's partnership with several leading cloud vendors, which also includes IBM and Google, finally gives the legions of enterprises that rely on VMware technology a way to migrate legacy workloads to the cloud.

Virtual Machines

AWS and Microsoft Azure deliver their core compute resources in the form of virtual machines—through Amazon EC2 and Azure Virtual Machines, respectively.

Both providers make it easy for you to find the right compute match for your needs by categorizing their resources by performance features and target use cases.

AWS does this by organizing them into instance types and instance families. Each instance type is a group of machines of different sizes, which combine CPU, memory, storage and network in the same proportion:

R3

R3 instances are optimized for memory-intensive applications and offer lower price per GiB of RAM.

Features:

- High Frequency Intel Xeon E5-2670 v2 (Ivy Bridge) Processors
- SSD Storage
- Support for Enhanced Networking

Model	vCPU	Mem (GiB)	SSD Storage (GB)
r3.large	2	15.25	1 x 32
r3.xlarge	4	30.5	1 x 80
r3.2xlarge	8	61	1 x 160
r3.4xlarge	16	122	1 x 320
r3.8xlarge	32	244	2 x 320

Amazon's R3 family of EC2 instances

An instance family is a group of instance types that share key defining characteristics, such as a high ratio of memory or low-latency directly attached storage:

Current Generation Instances

Instance Family	Current Generation Instance Types
General purpose	t2.nano | t2.micro | t2.small | t2.medium | t2.large | t2.xlarge | t2.2xlarge | m4.large | m4.xlarge | m4.2xlarge | m4.4xlarge | m4.10xlarge | m4.16xlarge | m3.medium | m3.large | m3.xlarge | m3.2xlarge
Compute optimized	c4.large | c4.xlarge | c4.2xlarge | c4.4xlarge | c4.8xlarge | c3.large | c3.xlarge | c3.2xlarge | c3.4xlarge | c3.8xlarge
Memory optimized	r3.large | r3.xlarge | r3.2xlarge | r3.4xlarge | r3.8xlarge | r4.large | r4.xlarge | r4.2xlarge | r4.4xlarge | r4.8xlarge | r4.16xlarge | x1.16xlarge | x1.32xlarge
Storage optimized	d2.xlarge | d2.2xlarge | d2.4xlarge | d2.8xlarge | i2.xlarge | i2.2xlarge | i2.4xlarge | i2.8xlarge | i3.large | i3.xlarge | i3.2xlarge | i3.4xlarge | i3.8xlarge | i3.16xlarge
Accelerated computing	f1.2xlarge | f1.16xlarge | p2.xlarge | p2.8xlarge | p2.16xlarge | g2.2xlarge | g2.8xlarge

Amazon's range of instance families

Microsoft uses a very similar system, but has adopted the term *series* as opposed to instance type:

ACU: 50-100

Size	CPU cores	Memory: GiB	Local HDD: GiB	Max data disks	Max data disk throughput: IOPS	Max NICs / Network bandwidth
Standard_A0*	1	0.768	20	1	1x500	2 / low
Standard_A1	1	1.75	70	2	2x500	2 / moderate
Standard_A2	2	3.5	135	4	4x500	2 / moderate
Standard_A3	4	7	285	8	8x500	2 / high
Standard_A4	8	14	605	16	16x500	4 / high
Standard_A5	2	14	135	4	4X500	2 / moderate
Standard_A6	4	28	285	8	8x500	2 / high
Standard_A7	8	56	605	16	16x500	4 / high

Microsoft's A-series of virtual machines

To define its wider groups of machines, Microsoft uses the term *type* as opposed to instance family:

Type	Sizes	Description
General purpose	DSv2, Dv2, DS, D, Av2, A0-7	Balanced CPU-to-memory ratio. Ideal for testing and development, small to medium databases, and low to medium traffic web servers.
Compute optimized	Fs, F	High CPU-to-memory ratio. Good for medium traffic web servers, network appliances, batch processes, and application servers.
Memory optimized	GS, G, DSv2, DS	High memory-to-core ratio. Great for relational database servers, medium to large caches, and in-memory analytics.
Storage optimized	Ls	High disk throughput and IO. Ideal for Big Data, SQL, and NoSQL databases.
GPU	NV, NC	Specialized virtual machines targeted for heavy graphic rendering and video editing. Available with single or multiple GPUs.
High performance compute	H, A8-11	Our fastest and most powerful CPU virtual machines with optional high-throughput network interfaces (RDMA).

Microsoft's range of virtual machine types

Each of the vendors' four main instance families or types are as follows:

General Purpose: A balanced mix of CPU and memory, suited to mainstream enterprise applications, testing and development, web servers and small to medium sized databases

Compute Optimized: A higher ratio of CPU to memory, designed for compute-intensive applications, such as medium to high traffic web servers, video encoding, and batch processing

Memory Optimized: A higher memory to CPU ratio, geared towards performance-sensitive databases and in-memory analytics

Storage Optimized: An emphasis on large amounts of directly attached storage, providing high I/O or throughput, intended for use with NoSQL databases and data warehouses

Hourly on-demand rates vary considerably and depend on a variety of factors, which include:

- **Type and size of virtual machine:** The higher the specification, the higher the cost.
- **AWS or Azure region in which you deploy your machine:** US regions are generally the least expensive.
- **Operating system (OS):** Windows and enterprise Linux distributions cost more than free versions of Linux, such as CentOS.

In addition to storage-optimized machines, a number of other instance types come with onboard ephemeral storage, which provides fast read and write times but only persists as long as your associated machine is running. However, if you use instances without local storage or need a more durable method of storing data, you'll need to provision persistent storage separately.

You can also deploy machines with preconfigured infrastructure or environments, such as **WordPress** or a **LAMP** or **ELK stack**, from either of the two vendors' respective marketplaces.

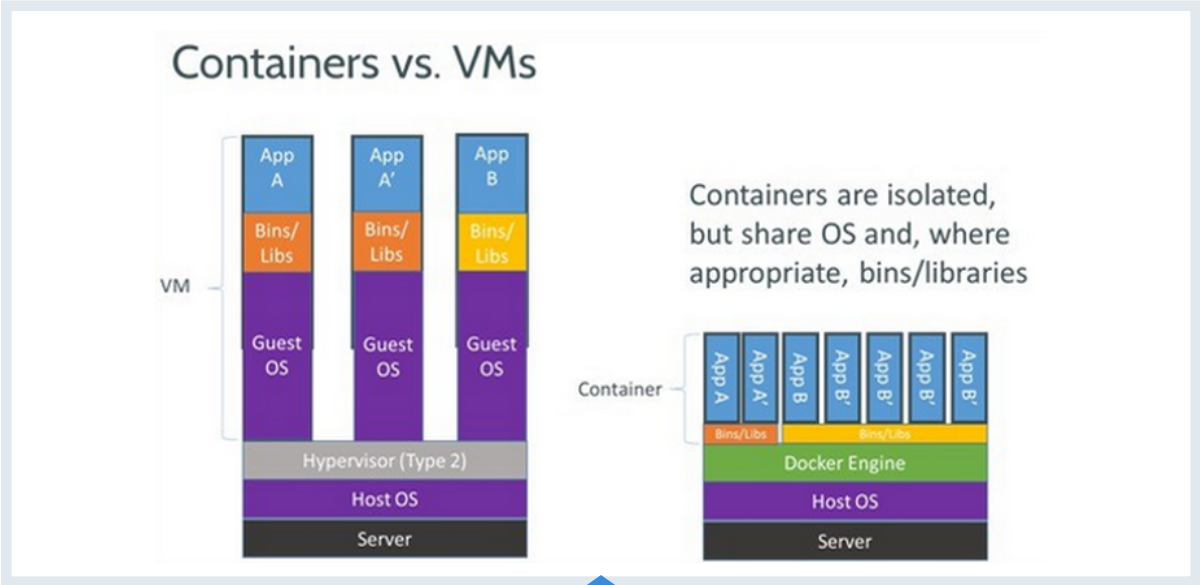
Finally, some of Amazon's managed database services, such as RDS and Redshift, also use the concept of instances to allocate compute resources. By contrast, Microsoft organizes its DBaaS compute resources into a tier system of performance levels.

Container Services

Container Basics

Containers are a relatively new application deployment technology and a streamlined alternative to virtual machines. They take advantage of the isolation features of the Linux kernel and share the resources of the host OS.

By contrast with a virtual machine, which requires a full OS to support your applications, a container has ready access to all the components it needs to run your code. This means you can make more efficient use of your compute resources and ultimately reduce your cloud costs.



Containers do away with the need for a hypervisor and sit at host OS level

Containers are used to host stateless applications and are designed for deployment to a cluster of instances as part of a distributed architecture of microservices—where each microservice is a group of tasks that run the same code in parallel. This method makes it easier to fine-tune your resource consumption, as you can:

- Scale horizontally by using a large number of smaller machines, thereby allowing you to add or remove resources from your cluster in smaller increments.
- Increase or decrease the number of tasks dedicated to the same microservice.

Containers can also play a valuable role as part of a hybrid cloud strategy, as they're highly portable and can be deployed to different servers with relative ease. And because they decouple the application from the underlying infrastructure, containers make life easier for the developer, freeing them up to focus on their code.

Open-source software Docker is the leading containerization platform. Although it was initially developed as a Linux technology, Microsoft recently introduced native support for Docker with the release of Windows Server 2016. This is set to widen usage to customers that run their systems on Windows OS, as well as increase workload mobility between different cloud and on-premise environments.

Container Offerings

Amazon's core container offering is EC2 Container Service (ECS). It is the vendor's own fully managed service, which makes it easy for customers to set up and maintain their container clusters. It also offers strong integration with other AWS services.

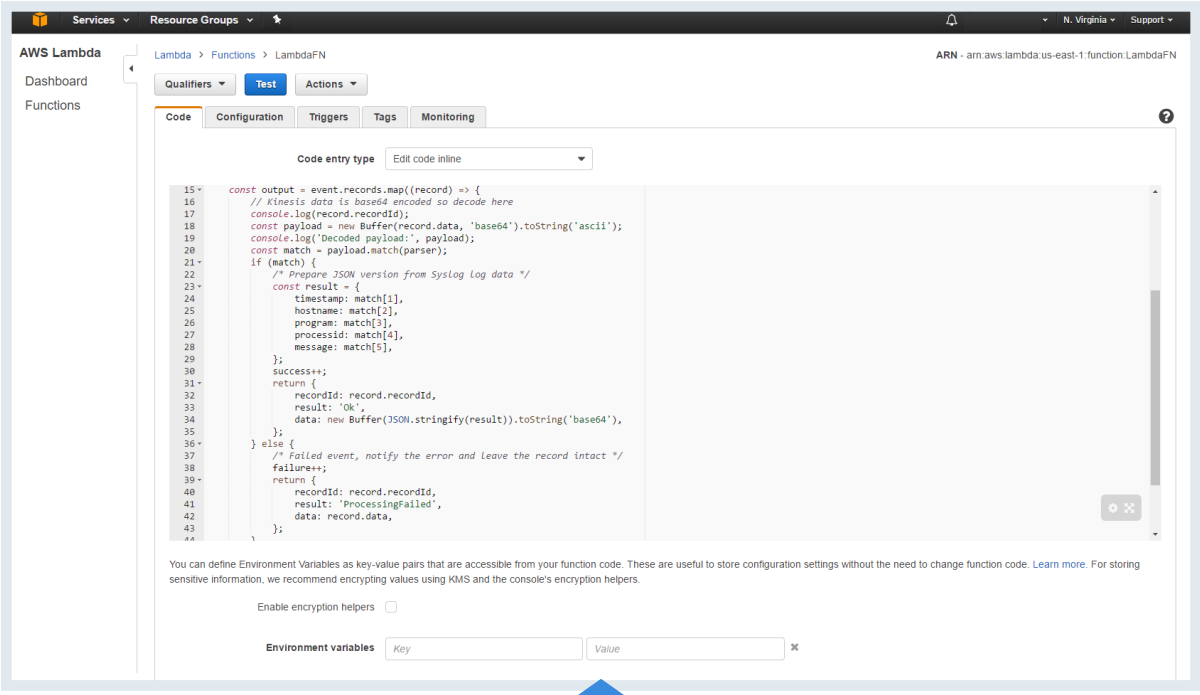
The vendor also supports several open-source container management engines, including Kubernetes, Mesos, and Docker Swarm. These solutions require more technical expertise to set up and manage. However, they offer more functionality and portability than Amazon's proprietary counterpart.

Microsoft's Azure Container Service (ACS) works slightly differently. It is built on a modular architecture that allows you to run the container orchestration tools of your choice. It currently supports Docker Swarm, Mesosphere DCOS and Kubernetes.

Serverless Solutions

Serverless Basics

Serverless computing takes the concepts of PaaS and container microservices to the next level by doing away with the construct of a server altogether. You simply submit your code and specify the events, such as changes to data, actions by users or API calls, that should trigger execution of the code.



AWS Lambda example

You don't need to worry about server administration or availability, as the vendor takes care of all this for you. And the only provisioning you need to do is allocate the amount of memory reservation to your code.

The key point about serverless solutions from a financial perspective is that you only pay for the compute time you actually consume. This can make them a cost-effective option for certain types of application, particularly microservices that may not make adequate use of a full-blown virtual machine.

Typical use cases of serverless computing include **log processing, ETL tasks, image conversion** and **IoT backend processing**.

Serverless Offerings

AWS Lambda

More than two years after its launch at AWS re:Invent 2014, Lambda has now emerged as one of Amazon's fastest-growing cloud services.

Language support: Node.js, Java, C# and Python

Monthly charges are based on the total number of requests + total memory consumption across all your functions, where:

- A request is counted each time your code starts to run in response to an event or execution call.
- Memory consumption = how much memory you allocate to a function x how long it is used.

AWS Lambda also includes a free tier, which gives you 1 million free requests and 400,000 GB-seconds of memory consumption each month.

Azure Functions

Azure Functions, which went on general release in November 2016, is still a relatively new service. However, it is available as both a serverless and traditional virtual machine solution, giving users more flexibility to optimize their costs.

Language support: JavaScript, C#, F#, Python and PHP

Pricing models	
Consumption Plan	Pricing works much like AWS Lambda, where you pay only when your application runs.
App Service Plan	Functions consume the resources of an existing provisioned virtual machine that comes under your App Service Plan. This can help make savings on your cloud bills, as the cost of running your functions is covered by the monthly charges for your virtual machine.

Discounted Compute Capacity

When assessing your compute options on AWS and Azure, it also pays to explore the various alternatives to the regular on-demand pricing model.

Both vendors offer payment plans, whereby you make a financial commitment to reserved compute capacity over a fixed period in exchange for a significant discount.

Amazon's **Reserved Instances (RIs)** and the Azure **Compute Pre-Purchase Plan (CPP)** both work on a credit system, which entitles you to free usage of the machine type you specify throughout the term of the plan.

RIs offer a number of purchase options—all upfront, partial upfront or no upfront payment at all—along with terms of either one or three years. In terms of payment options, the CPP is more basic and is only available on a one-year term. However, it does offer a more flexible approach to credit allocation.

Download the Buyer's Guide to AWS Reserved Instances

Whereas an RI credit can only apply to one matching instance at any one time, the CPP can apply to any number of concurrently running machines that match your specification. This can help you get more utilization from your usage entitlement of 744 hours each month.

Amazon also offers Spot Instances, which allow you to bid on excess EC2 capacity at a reduced cost. Spot Instances are a great fit for large-scale batch processing jobs, such as **media coding, security testing** and **insurance risk management**, and can offer significant savings potential. However, they require proactive management and optimization to be efficient for everyday mission-critical services that rely on continuous availability.

Conclusion

With a clear idea of the compute options available, you'll be better equipped to calculate the cost of moving your applications and make a successful migration to the cloud.

But compute is only the first step in the cost evaluation process. You'll also need to include other services in your projected costs, such as load balancing, storage, content delivery networks, data transfer and any additional infrastructure required for a fault-tolerant environment.

As well as significant potential savings compared with traditional on-premise infrastructure, the cloud also offers a modern, flexible platform for your applications. So don't forget to consider the wider benefits of hosting in the cloud, such as fast provisioning, auto-scaling, and automation.

It's critical to truly understand the cloud's pay-as-you-go infrastructure to ensure your organization has measures in place to optimize costs. Proactive resource sizing, comprehensive infrastructure monitoring, and cost-aware cloud architecture can enable you to make the most of your cloud investment.

About CloudCheckr

The CloudCheckr platform offers a single pane of glass across infrastructure to ensure total security and compliance, while optimizing cost and expenses. With continuous monitoring, 450 best practice checks, and built-in automation, CloudCheckr helps organizations to ensure compliance for highly regulated industries, with alerts, monitoring, and audits to meet FedRAMP, DFARS, HIPAA, PCI, and other security standards. With deeper intelligence across cloud infrastructure and a unified cloud management solution, organizations can prevent risks and mitigate threats before they occur. Get started at cloudcheckr.com/getstarted.

VISIT US ONLINE