**Chapter 1 DESCRIBE CLOUD CONCEPTS**

**CAPEX** is the spending of money on physical infrastructure upfront and then deducting that expense from your tax bill over time. CapEx is an **upfront cost**, which has a value that reduces over time and usually has **no recurring cost**.  
  
**OpEx** is your operating costs, the expenses to run the day-to-day business, like services and consumable items that get used up and are paid for according to use. You can deduct this expense from your tax bill in the same year. There’s**no upfront cost**but has a **recurring cost**. You pay for a service or product as you use it i.e. **pay-as-you-go pricing.**

**Network outage  
Application failure**: Application Insights that integrates with your application to give you detailed information about the performance and reliability of your application. Application developers can often use this information to get right to the code where a problem is happening, dramatically reducing the time needed for troubleshooting.  
  
**System outage  
Power outage**: Cloud providers invest heavily in battery operated power backups and other redundant systems in order to prevent availability problems caused by power outages.  
In a situation where a large geographic area is affected by a power outage, cloud providers offer you the ability to run your application from another region that isn’t affected

**Scaling and elasticity**  
Scaling is the process of adding additional resources or additional power for your application.   
**There are two variations of scaling:**   
**Horizontal scaling** (*often referred to as scaling out*) - You add additional VMs for your application. Each VM you add is identical to other VMs servicing your application.  
**Vertical scaling** (often referred to as scaling up) - Scaling out provides additional resources to handle additional load. *For example, you might determine that you need a more powerful CPU and more memory for your application. In that case, scaling up will allow you to move your application to a more powerful VM.*

**The concept of automatically scaling is referred to as elasticity.  
Speed and flexibility in the cloud is often called cloud agility.**  
  
**Don’t confuse fault tolerance with scaling:**Scaling allows you to react to additional load or resource needs, but it’s always assumed that **all the VMs you are using are healthy**.   
**Fault tolerance** happens without any interaction from you, and **it’s designed to automatically move you from an unhealthy system to a healthy system if things go wrong**  
  
**Disaster Recovery** **and** **Governments**   
Depending on what kind of data you store, you might be required to have a disaster recovery plan in place. Cloud providers typically comply with standards imposed by laws such as the **Health Insurance Portability and Accountability Act** **(HIPAA),** and they often provide compliance tools you can use to ensure compliance.  
**Disaster recovery** not only means having **reliable backups of important data**, but it also means that the **cloud infrastructure can replicate your application’s resources in an unaffected region** so that your data is safe and your application availability isn’t affected.   
**Disaster recovery** plans are commonly referred to as **Business Continuity and Disaster Recovery (BCDR)** plans, and most cloud providers have services that can help you develop and implement a plan that works for your particular needs

**SKILL 1.2: DESCRIBE THE DIFFERENCES BETWEEN INFRASTRUCTURE-AS-ASERVICE (IAAS), PLATFORMAS-A-SERVICE (PAAS), AND SOFTWARE-AS-A-SERVICE (SAAS)**

**IAAS**  
Once you have an IaaS VM running in the cloud, you gain access to many services the cloud provider offers. For example, Microsoft offers **Azure Security Center** to ensure the security of your IaaS VMs, **Azure Backup** to make backing up data easy, **Azure Log Analytics** to help with troubleshooting any problems you might have, and much more  
IaaS is also a great choice if you want your application and configuration in the cloud, but you want the option of not paying for it when you aren’t using it. By stopping your VM, you can avoid the costs associated with it, and when you need to use your application again, you can simply start your VM and pick up right where you left off.

**PAAS**  
If you are deploying your own application to the cloud and you want to minimize your management investment, a PaaS service is often the best choice.  
A PaaS service also uses VMs provided by the cloud provider. However, a user typically has no visibility into those VMs. In most cases, they’re entirely managed by the cloud provider.

* **Azure App Service** one of the PaaS offerings in Azure. It has been created on a VM that’s maintained by Microsoft. Notice the option to choose either Linux or Windows, but the operating system is still managed by Microsoft. We also have the option to enable Application Insights, a service in Azure that provides deep insight into how an application is performing, making it easier to troubleshoot problems if they occur.

In Azure App Service, you don’t have to worry about Docker installation or configuration. It’s automatically included on all App Service VMs as part of Microsoft’s PaaS offering, and it’s completely managed and maintained by Microsoft. Examples: **Azure CDN, Azure Cosmos DB, Azure SQL Database, Azure Database for MySQL, Azure Storage, Azure Synapse Analytics**

**Lift-and-shift** It’s a concept that you can often move your application from on-premises to a cloud environment by simply deploying it to the cloud

**SAAS**  
SaaS service is software provided by a cloud provider that’s installed on infrastructure completely controlled by the hosting provider. Examples: Microsoft 365, Xbox Live & One Drive.

**SKILL 1.3: DESCRIBE THE DIFFERENCES BETWEEN PUBLIC, PRIVATE, AND HYBRID CLOUD MODELS**

**The public cloud**  
The most common cloud model is the public cloud. In a public cloud model, you use shared infrastructure that is accessible on a public network. The network, storage, and VMs that your application uses are provided by a cloud provider and shared between all consumers of the public cloud. Microsoft Azure, Amazon Web Services (AWS), and Google Cloud Platform are examples of public clouds.

**The private cloud**  
The private cloud model provides many of the attractive benefits of the cloud (things like easy scaling and elasticity) in a private environment that is dedicated to a single company. A private cloud can be hosted in an on-premises environment, but it can also be hosted on a thirdparty hosting provider.

**The hybrid cloud**  
As you might expect, hybrid clouds are a mixture of public and private clouds.

**Chapter 2 DESCRIBE CORE AZURE SERVICES**

**Azure regions**   
Within the United States geography, there are many regions, including the Central US region in Iowa, the East US region in Virginia, the West US region in California, and the South Central US region in Texas.  
Microsoft also operates isolated regions that are completely dedicated to government data because of the additional regulations that governmental data requires

**Regional pair  
E**ach regional pair contains two regions within the geography. **When Microsoft has to perform updates to the Azure platform, they perform those updates on one region in the regional pair. Once those updates are complete, they move to the next region in the regional pair. This ensures that your services operating within a regional pair aren’t impacted by updates.**

The fact that each **geography contains at least two regions separated by a large physical distance is important. That’s how Azure maintains disaster recovery.**

*When a customer is creating Azure resources, only the region is visible.* ***The concept of geographies is an internal implementation of Azure*** *that customers don’t really have visibility of when using Azure.***Customers also don’t have visibility into the concept of regional pairs, but they can see each region within a regional pair.**

In order to ensure that applications are still performing as quickly as possible, **Microsoft guarantees round-trip network performance of 2 milliseconds or less between regions.**

**Availability zones**  
Important that data and applications maintain availability **when a problem occurs at a particular datacenter within a region**. For that reason, Microsoft developed availability zones.

Currently, availability zones are supported with the following Azure services: <https://docs.microsoft.com/en-us/azure/availability-zones/az-region>

Don’t confuse availability zones with availability sets.  
**Availability sets**   
Allow you to create two or more virtual machines in different physical server racks in an Azure datacenter. Microsoft guarantees a **99.95 percent SLA with an availability set.   
Availability zone**  
Allows you to deploy **two or more Azure services into two distinct datacenters within a region.** Microsoft guarantees a **99.99 percent SLA with availability zones.**

**Zonal services** – A resource can be deployed to a specific, self-selected Availability Zone to achieve more stringent latency or performance requirements. Resiliency is self-architected by replicating applications and data to one or more zones within the region. Resources can be pinned to a specific zone. For example, virtual machines, managed disks, or standard IP addresses can be pinned to a specific zone, which allows for increased resilience by having one or more instances of resources spread across zones.

**Zone-redundant services** – Azure platform replicates the resource and data across zones. Microsoft manages the delivery of high availability since Azure automatically replicates and distributes instances within the region. ZRS, for example, replicates the data across three zones so that a zone failure does not impact the HA of the data.

**Non-regional services** – Services are always available from Azure geographies and are resilient to zone-wide outages as well as region-wide outages.  
 **Resource group**If you have access to multiple Azure subscriptions, you can also have resources from multiple subscriptions in a single resource group.

An Azure resource can only exist in one resource group. In other words, you can’t have a virtual machine in a resource group called WebStorefront and also in a resource group called SalesMarketing, because **it must be in one group or the other. You can move Azure resources from one resource group to another**

**Azure subscriptions**Each Azure subscription has limits (sometimes called quotas) assigned to it.   
For example, you can have up to **250 Azure Storage accounts per region in a subscription**, **up to 25,000 virtual machines per region, and up to 980 resource groups per subscription across all regions.**

*Microsoft support can increase limits in some scenarios if you have a good business justification. Some limits, however, cannot be increased.*

**Free Trial Provides** free access to Azure resources for a limited time.

**Pay-As-You-Go** You pay only for those resources you use in Azure. There’s no up-front cost, and you can cancel the subscription at any time.

**Pay-As-You-Go Dev. / Test** A special subscription for subscribers to Visual Studio that can be used for development and testing. This subscription offers discounted rates on VMs, but you cannot use this for production applications.

*Each subscription is associated with a unique identifier called a* ***subscription ID****. You can give each subscription a descriptive name to help you identify it, but Azure will always use the subscription ID to identify your subscription. When you talk to Microsoft about your Azure account, they’ll also often ask for your subscription ID.*

**Management groups  
A convenient way to apply policies and access control to your Azure resources.** Much like a resource group, a management group **is a container for organizing your resources**. However, management groups can contain only **Azure subscriptions or other management groups**.

Diagram, timeline

Description automatically generated

*By organizing the subscriptions using management groups, you can have more precise control over who has access to which resources. You can also control the configuration of resources created within those subscriptions.*

There are, however, a few limitations: **You’re limited to a total of 10,000 management groups**. A management group hierarchy can only **support up to six levels**. You cannot have multiple parents for a single management group or subscription.

**Azure Resource Manager (ARM)**In order to make it easier to deploy and manage Azure services, Microsoft developed Azure Resource Manager, or ARM. **ARM is a service that runs in Azure, and it’s responsible for all interaction with Azure services.** When you create a new Azure service, ARM authenticates you to make sure you have the right access to create that resource, and then it talks to a resource provider for the service you’re creating. For example, **if you’re creating a new web app in Azure App Service, ARM will pass your request on to the Microsoft.Web resource provider because it knows all about web apps and how to create them.**

Timeline

Description automatically generated

**You don’t have to tell ARM how to do what you want. You simply have to tell it what you want.** To do that, ARM uses files that are encoded in JavaScript Object Notation (or JSON) called ARM templates.

ARM allows you to specify dependencies so you can avoid issues like this.

* **ARM allows you to easily deploy multiple Azure resources at once.**
* ARM makes it possible to reproduce any deployment with consistent results at any point in the future.
* ARM allows you to create declarative templates for deployment instead of requiring you to write and maintain complex deployment scripts.
* **ARM makes it possible to set up dependencies so that your resources are deployed in the right order every time.**

**SKILL 2.2: DESCRIBE CORE WORKLOAD PRODUCTS AVAILABLE IN AZURE**

**Azure virtual machines**   
Because the VMs running on a host use the physical systems on that host, if you have a need for a powerful VM, you’ll need a powerful physical computer to host it.

When you click Create to create your VM, the Azure portal is actually using an ARM template to deploy your VM. That ARM template contains parameters that are replaced with the information you entered for your VM. Every VM that is created in Azure is created using an ARM template. This ensures that the deployments are consistent.

**Keep in mind that unless you have configured a static IP address for your VM, your IP address will likely change the next time you start it.**

**You can also stop a VM from within the guest operating system on the VM, but when you do that, you will still be charged for the resources the VM uses because it’s still allocated to you.**

**Planned maintenance**   
Includes things like operating system updates, driver updates, and so on.

Azure has underlying systems that constantly monitor the health of computer components. If one of these underlying systems detects that a component within the host computer might fail soon, Azure will flag the computer for unplanned maintenance.   
  
**Unplanned maintenance**Azure will attempt to move your VM to a healthy host computer. When it does this, it preserves the state of the VM, including what’s in memory and any files that are open. It only takes Azure a short time to move the VM, during which time it’s in a paused state. In a case where the move operation fails, the VM will experience **unexpected downtime**.

*In order to ensure reliability when a failure occurs in a rack within the Azure datacenter, you can (and you should) take advantage of a feature called availability sets.* **Availability sets** *protect you from maintenance events and downtime caused by hardware failures. To do that, Azure creates some underlying entities in an availability set called update domains and fault domains. (In order to protect yourself in the event of maintenance events or downtime, you must deploy at least two VMs into your availability set.)*

**Fault domains**   
**A logical representation of the physical rack in which a host computer is installed**. By default, Azure assigns two fault domains to an availability set. If a problem occurs in one fault domain (one computer rack), the VMs in that fault domain will be affected, but VMs in the second fault domain will not. **This protects you from unplanned maintenance events and unexpected downtime.**

**Update domains**   
Designed to protect you from a situation where the host computer is being rebooted. **When you create an availability set, Azure creates five update domains by default**. These update domains are spread across the fault domains in the availability set.   
If a reboot is required on computers in the availability set (whether host computers or VMs within the availability set), **Azure will only reboot computers in one update domain at a time and it will wait 30 minutes for computers to recover from the reboot before it moves on to the next update domain. Update domains protect you from planned maintenance events.**

Graphical user interface, diagram

Description automatically generated

Graphical user interface

Description automatically generated

*If your availability set is servicing a website hosted on the VMs, you’ll need to configure a load balancer that will handle the job of routing users of your website to the VMs that are running it.*

**Scale set**When you create a scale set, **you tell Azure what operating system you want to run and then you tell Azure how many VMs you want in your scale set**. You have many other options such as creating a load balancer or gateway and so forth. Azure will create as many VMs as you specify (up to 1,000) in one easy step.

***Scale sets provide that functionality by using Azure’s auto-scale feature. You define scaling rules that use metrics like CPU, disk usage, network usage, and so forth.*** You can configure when Azure should add additional instances and when it should scale back and deallocate  
*VMs in a scale set are also compatible with availability zones, so you are protected from problems in an Azure datacenter.*

**App Service plans**Every web app you create in App Service runs inside of an App Service plan. An App Serviceplan is created within a specific Azure region, and it specifies how many VMs your app runs on and the properties of those VMs.  
Depending on the tier of service you use when you create your app, it will either run on a VM that is shared among many users or a VM that is dedicated to you.

**Multiple apps can run inside of a single App Service plan. All apps in an App Service plan will share the same VMs in that App Service plan.**

Graphical user interface, application

Description automatically generated

* **Free** A no-cost tier for testing only that runs on VMs shared with other App Service customers.
* **Shared** A low-cost tier for testing only with some additional features not offered in the Free tier. Runs on VMs shared with other App Service customers.
* **Basic, Standard, Premium, and PremiumV2** Highercost tiers that offer many additional features. Runs on dedicated VMs that are not shared with other customers.

**You are charged for App Service plans even when no web apps are running in them. If you do have web apps in your App Service plan, you are still charged if you stop the web apps. The only way to avoid being billed for an App Service plan is to delete it.**

*Creating a web app in App Service is very fast and scaling it out to multiple instances is also very fast. That’s because the VMs that are running App Service web apps are already up and running. When you create a web app, you are simply allocating an existing VM for your use.*

**App Service**   
Allows you to choose between a VM preconfigured with a runtime stack (such as Java, .NET, PHP, and so forth) to run your app or a Docker container.

**You can see many of the features available in App Service, including the ability to quickly and easily scale out when needed.**  
Graphical user interface, text, application, email

Description automatically generated

**Azure Container Instances (ACI)  
Azure creates server resources as needed to run your container, but you’re not paying for an underlying VM. Instead, you pay for the memory and CPU that your container uses.**

***ACI*** *is designed to work with simple applications. You can define a container group and run multiple containers within an ACI instance, but ACI isn’t a good choice for you if you have an application that is used heavily by many people and that might need to take advantage of scaling. Instead, Azure Kubernetes Service (AKS) would be a better choice.  
  
You can’t change the DNS Name Label after the instance is created. You also can’t change the image your instance uses. If you want to change these settings, you’ll need to delete the instance and re-create it. However, doing so might mean that you lose your public IP address, so it’s best to plan ahead before you create your instance.*

**Azure Kubernetes Service (AKS)**Kubernetes creates containers in a pod. A pod is a group of related containers, and containers within a pod can share resources.  
However, a container in one pod is not able to share resources with a container in another pod.  
The computer that Kubernetes pods are running on is called a node or a worker.  
In addition to pods, the node also runs several services that are required for Kubernetes to manage the pods and so on.

There will typically be multiple nodes within a Kubernetes instance, and they are all controlled by a master node called the Kubernetes master. The entire environment of the master and all its nodes is called a **Kubernetes cluster.**

When you create a Kubernetes cluster in AKS, Azure creates the master and the nodes for you. All you have to do is deploy your containers and you’re up and running with a managed Kubernetes cluster.

**AKS in Azure is free. You only pay for the Azure compute resources you use within your cluster**