DigitalCloud TRAINING

What is the Exam Cram PDF and how to use it:

- These are basically notes taken for you - a summary of key facts that you need to know to pass the exam
- Use these for quick revision, and please feel free to print them out
- All of these topics are explained in more detail in the video lessons
- All of these topics (and more) are documented in the Training Notes on our website (and which you can also download as a PDF file)
- The short quizzes at the end of the section relate to topics that were covered

Section 2: Cloud Terminology

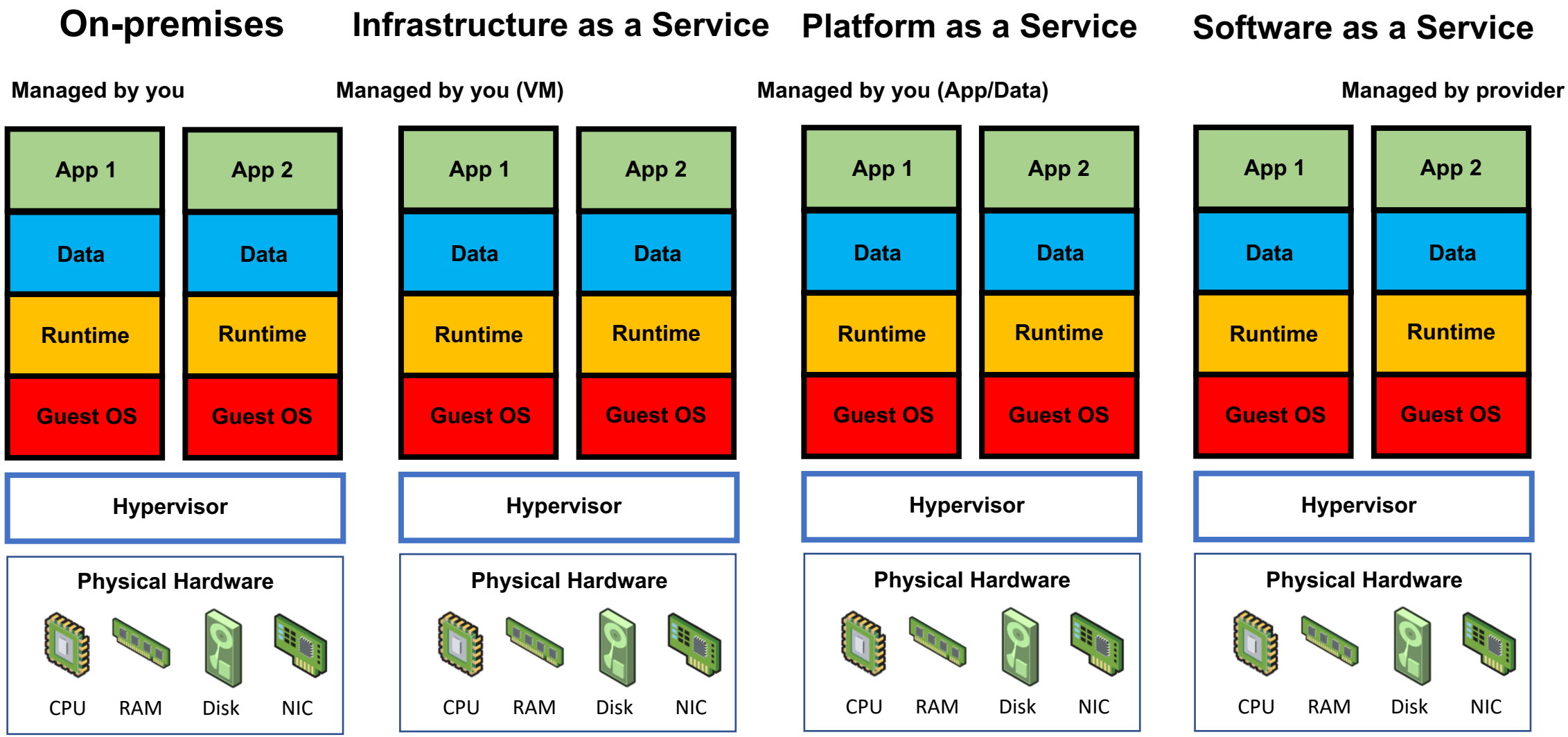
DigitalCloud TRAINING

Term	Description
Cloud Computing	Cloud computing is the on-demand delivery of IT services from a third-party provider over the Internet
Cloud Service	The IT capability that is being provided by the cloud provider
Cloud Provider / Cloud Service Provider	A company that provides a cloud service to organizations and/or individuals
Consumer	The organization or individual who uses the cloud service
”Pay as you go” or “pay per use”	You are charged only for what you use. Analogous to a utility bill
Multi-tenant	Multiple customers consume services delivered using shared infrastructure
“x” as a service	Some cloud capability is delivered to consumers as a service

Section 2: Key Characteristics of Cloud Computing

DigitalCloud TRAINING

Name	Description
On-demand, self-service	**A user can consume cloud resources, as needed, automatically, and without human interaction**
Broad network access	**Capabilities are available over the network using standard mechanisms. Can be the Internet or a Wide Area Network (WAN)**
Resource pooling	**The providers resources are pooled and serve multiple consumers using a multi-tenant model**
Rapid elasticity	**Capabilities can scale “elastically” based on demand**
Measured service	**Resource usage is monitored and metered**


On-premises
Managed by you
App 1
App 2
Data
Data
Runtime
Runtime
Guest OS
Guest OS
Hypervisor
Physical Hardware
CPU
RAM
Disk
NIC
Infrastructure as a Service
Managed by you (VM)
App 1
App 2
Data
Data
Runtime
Runtime
Guest OS
Guest OS
Hypervisor
Physical Hardware
CPU
RAM
Disk
NIC
Platform as a Service
Managed by you (App/Data)
App 1
App 2
Data
Data
Runtime
Runtime
Guest OS
Guest OS
Hypervisor
Physical Hardware
CPU
RAM
Disk
NIC
Software as a Service
Managed by provider
App 1
App 2
Data
Data
Runtime
Runtime
Guest OS
Guest OS
Hypervisor
Physical Hardware
CPU
RAM
Disk
NIC

Name	Description	Examples
Private Cloud	An enterprise deploys their own infrastructure and applications into their own data center	VMware, Microsoft, RedHat, OpenStack
Public Cloud	The IT services that you consume are hosted and delivered from a third-party and accessed over the Internet	AWS, Microsoft Azure, Google Cloud Platform
Hybrid Cloud	A combination of on-premises, private cloud, and public cloud services are consumed	
Multicloud	Usage of two or more public clouds at a time, and possibly multiple private clouds	

Section 2: The 6 Advantages of Cloud

	Name	Description
1	Trade capital expense for variable expense	Instead of investing in data centers before you know how you're going to use them, pay only when, and for how much, you consume
2	Benefit from massive economies of scale	Achieve a lower variable cost due to AWS' scale
3	Stop guessing about capacity	Eliminate guessing, scale as demand dictates
4	Increase speed and agility	Easily and quickly scale your usage
5	Stop spending money running and maintaining data centers	Focus on business growth and innovation instead!
6	Go global in minutes	Easily deploy applications in multiple regions around the world

Name	Description
Region	A geographical area with 2 or more AZs, isolated from other AWS regions
Availability Zone (AZ)	One or more data centers that are physically separate and isolated from other AZs
Edge Location	A location with a cache of content that can be delivered at low latency to users – used by CloudFront
Regional Edge Cache	Also part of the CloudFront network. These are larger caches that sit between AWS services and Edge Locations
Global Network	Highly available, low-latency private global network interconnecting every data center, AZ, and AWS region

Section 3: Global vs Regional Services

DigitalCloud TRAINING

Identity and Access Management

- AWS IAM

AWS Compute

- Amazon EC2
- Amazon ECS
- AWS Lambda
- Amazon LightSail

AWS Storage

- Amazon S3
- Amazon EBS
- Amazon EFS
- AWS Storage Gateway

AWS Networking

- Amazon VPC
- AWS Direct Connect

Databases

- Amazon RDS
- Amazon DynamoDB
- Amazon RedShift
- Amazon ElastiCache

Elastic Load Balancing and Auto Scaling

- Elastic Load Balancing
- Auto Scaling

Content Delivery and DNS Services

- Amazon Route 53
- Amazon CloudFront

Monitoring and Logging Services

- Amazon CloudWatch
- AWS CloudTrail

Notification Services

- Amazon SNS

Migration and Transfer

- Database Migration Service
- Server Migration Service
- AWS Snowball

Cloud Governance and Security

- AWS GuardDuty
- AWS KMS
- AWS WAF & Shield
- AWS CloudHSM
- AWS Artifact
- AWS Inspector
- AWS Trusted Advisor
- AWS Config
- AWS Service Catalog
- AWS Personal Health Dashboard

- An AWS region is a geographical area
- Each region consists of 2 or more availability zones
- Each Amazon Region is designed to be completely isolated from the other Amazon Regions
- Availability Zones (AZs) are locations into which you launch resources, such as Amazon EC2 instances
- AZs are physically separate and isolated from each other
- AZs span one or more data centers and have direct, low-latency, high throughput and redundant network connections between each other
- Each AZ is designed as an independent failure zone
- AZs are physically separated within a typical metropolitan region, and use discrete power sources

- The three fundamentals of AWS pricing are Compute, Storage and outbound data transfer
- On-demand
  - Used for Compute and Database capacity
  - No long-term commitments or upfront payments
- Dedicated Instances
  - Available for Amazon EC2
  - Hardware is dedicated to a single customer
- Spot Instances
  - Purchase spare capacity with no commitments
  - Great discounts from hourly rates
- Reservations
  - Up to 75% discount in exchange a term commitment

- Options for 1 or 3 year term
- Options to pay:
  - No upfront
  - Partial upfront
  - All upfront
- Available for these services:
  - Amazon EC2 Reserved Instances
  - Amazon DynamoDB Reserved Capacity
  - Amazon ElastiCache Reserved Nodes
  - Amazon Relational Database Service (RDS) Reserved Instances
  - Amazon RedShift Reserved Nodes
- AWS Acceptable Use Policy describes the prohibited uses of AWS

IAM Users

- An IAM user is an entity that represents a person or service
- Can be assigned:
  - An access key ID and secret access key for programmatic access to the AWS API, CLI, SDK, and other development tools
  - A password for access to the management console
- By default users cannot access anything in your account
- The account root user credentials are the email address used to create the account and a password
- The root account has full administrative permissions and these cannot be restricted

DigitalCloud TRAINING

IAM Groups

- Groups are collections of users and have policies attached to them
- A group is not an identity and cannot be identified as a principal in an IAM policy
- Use groups to assign permissions to users
- Use the principal of least privilege when assigning permissions
- You cannot nest groups (groups within groups)

IAM Roles

- Roles are created and then “assumed” by trusted entities and define a set of permissions for making AWS service requests
- With IAM Roles you can delegate permissions to resources for users and services without using permanent credentials (e.g. user name and password)

IAM Policies

- Policies are documents that define permissions and can be applied to users, groups and roles
- Policy documents are written in JSON (key value pair that consists of an attribute and a value)
- All permissions are implicitly denied by default
- The most restrictive policy is applied

Authentication Methods

- There are three authentication methods: Access Keys, Console Password and Server/Signing Certificates
- Access keys:
  - A combination of an access key ID and a secret access key
  - Used to make programmatic calls to AWS using the API or CLI
  - The secret access is returned only at creation time

Authentication Methods contd.

- Console Password
  - A password that the user can enter to sign in to interactive sessions such as the AWS Management Console
  - You can allow users to change their own passwords
- Server/Signing Certificate
  - SSL/TLS certificates that you can use to authenticate with some AWS services

Multi-Factor Authentication (MFA)

- 2-factors used with AWS: something you “know” – a password, and something you ”have” – a virtual or physical MFA device
- Adds extra security to your account

AWS Security Token Service (STS)

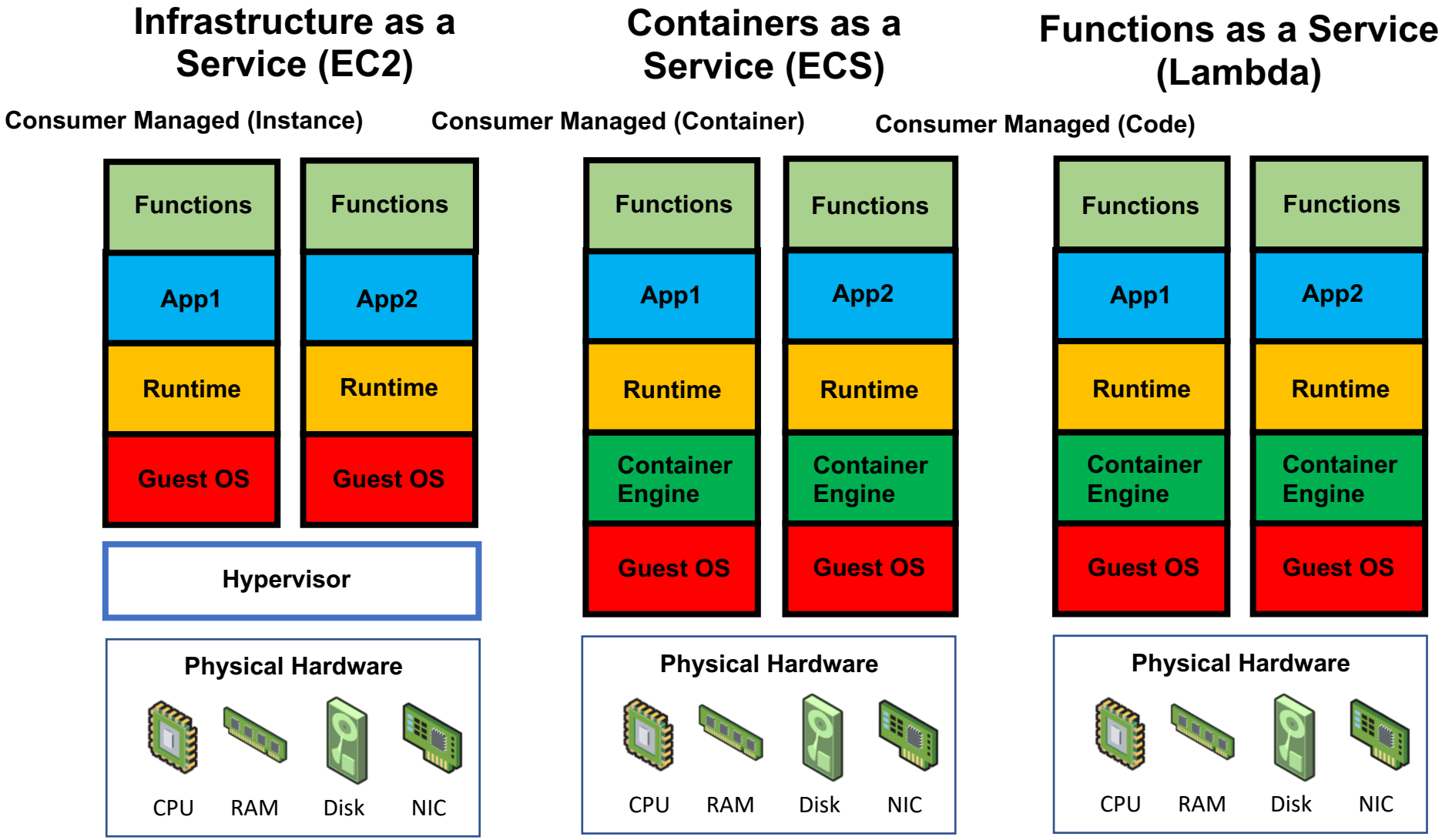
- The AWS Security Token Service (STS) is a web service that enables you to request temporary, limited-privilege credentials

IAM Best Practices

- Lock away the AWS root user access keys
- Create individual IAM users
- Use AWS defined policies to assign permissions whenever possible
- Use groups to assign permissions to IAM users
- Grant least privilege
- Use access levels to review IAM permissions
- Configure a strong password policy for users
- Enable MFA for privileged users
- Use roles for applications that run on AWS EC2 instances
- Delegate by using roles instead of sharing credentials
- Rotate credentials regularly
- Remove unnecessary credentials
- Use policy conditions for extra security
- Monitor activity in your AWS account

Section 6: ECS Terminology

Elastic Container Service (ECS)	Description
Cluster	Logical grouping of EC2 instances
Container instance	EC2 instance running the the ECS agent
Task Definition	Blueprint that describes how a docker container should launch
Task	A running container using settings in a Task Definition
Service	Defines long running tasks – can control task count with Auto Scaling and attach an ELB


Infrastructure as a Service (EC2)
Containers as a Service (ECS)
Functions as a Service (Lambda)
Consumer Managed (Instance)
Consumer Managed (Container)
Consumer Managed (Code)
Functions
App1
Runtime
Guest OS
Functions
App2
Runtime
Guest OS
Hypervisor
Physical Hardware
CPU
RAM
Disk
NIC
Functions
App1
Runtime
Container Engine
Guest OS
Functions
App2
Runtime
Container Engine
Guest OS
Physical Hardware
CPU
RAM
Disk
NIC
Functions
App1
Runtime
Container Engine
Guest OS
Functions
App2
Runtime
Container Engine
Guest OS
Physical Hardware
CPU
RAM
Disk
NIC

EC2	ECS (EC2 Launch Type)	ECS (Fargate Launch Type)	Lambda
You manage the operating system	You manage container instance (EC2) and the containers (tasks)	You manage the containers (tasks)	You manage the code
Scale vertically – more CPU/Mem/HDD or scale horizontally (automatic) with Auto Scaling	Manually add container instances or use ECS Services and EC2 Auto Scaling	AWS scales the cluster automatically	Lambda automatically scales concurrent executions up to default limit (1000)
Use for traditional applications and long running tasks	Use for microservices and batch use cases where you need containers and need to retain management of underlying platform	Use for microservices and batch use cases	Use for ETL, infrastructure automation, data validation, mobile backends
Pay for instance run time based on family/type	Pay for instance run time based on family/type	Pay for container run time based on allocated resources	Pay only for execution time based on memory allocation

DigitalCloud TRAINING

Amazon EC2

- Amazon Elastic Compute Cloud (Amazon EC2) is a web service in the AWS Compute suite of products that provides secure, resizable compute capacity in the cloud
- Elastic Web-Scale computing – you can increase or decrease capacity within minutes and commission one to thousands of instances simultaneously
- Completely controlled – You have complete control include root access to each instance and can stop and start instances without losing data and using web service APIs
- Flexible Cloud Hosting Services – you can choose from multiple instance types, operating systems, and software packages as well as instances with varying memory, CPU and storage configurations
- You select an instance family and type depending on your application requirements – this determines the amount of CPU, RAM, disk, and network throughput your instance has

Amazon Machine Image (AMI)

- An Amazon Machine Image (AMI) provides the information required to launch an instance
- An AMI includes the following:
  - One or more EBS snapshots, or, for instance-store-backed AMIs, a template for the root volume of the instance (for example, an operating system, an application server, and applications).
  - Launch permissions that control which AWS accounts can use the AMI to launch instances.
  - A block device mapping that specifies the volumes to attach to the instance when it's launched
- AMIs come in three main categories:
  - Community AMIs - free to use, generally you just select the operating system you want
  - AWS Marketplace AMIs - pay to use, generally come packaged with additional, licensed software
  - My AMIs - AMIs that you create yourself

User Data

- User data is data that is supplied by the user at instance launch in the form of a script
- User data is limited to 16KB
- User data and metadata are not encrypted

Metadata

- Instance metadata is data about your instance that you can use to configure or manage the running instance
- Instance metadata is available at http://169.254.169.254/latest/meta-data
- The Instance Metadata Query tool allows you to query the instance metadata without having to type out the full URI or category names

Amazon ECS

- Amazon Elastic Container Service (ECS) provides a highly scalable, high performance container management service that supports Docker containers
- A container is similar to a virtual instance, but there's less to manage
- With containers the code, runtime, system tools, system libraries and settings are packaged up
- Containers run quickly and reliably from one computing environment to another
- Amazon ECS eliminates the need for you to install, operate, and scale your own cluster management infrastructure

- An Amazon ECS launch type determines the type of infrastructure on which your tasks and services are hosted
- There are two launch types and the table below describes some of the differences between the two launch types:

Amazon EC2	Amazon Fargate
You explicitly provision EC2 instances	The control plane asks for resources and Fargate automatically provisions
You're responsible for upgrading, patching, care of EC2 pool	Fargate provisions compute as needed
You must handle cluster optimization	Fargate handles cluster optimization
More granular control over infrastructure	Limited control, as infrastructure is automated

DigitalCloud TRAINING

AWS Lambda

- AWS Lambda is a serverless computing technology that allows you to run code without provisioning or managing servers
- AWS Lambda executes code only when needed and scales automatically
- You pay only for the compute time you consume (you pay nothing when your code is not running)
- Benefits of AWS Lambda:
  - No servers to manage
  - Continuous scaling
  - Subsecond metering
  - Integrates with almost all other AWS services

Amazon Lightsail

- Amazon LightSail is great for users who do not have deep AWS technical expertise as it make it very easy to provision compute services
- Amazon Lightsail provides developers compute, storage, and networking capacity and capabilities to deploy and manage websites, web applications, and databases in the cloud
- Amazon Lightsail includes everything you need to launch your project quickly – a virtual machine, SSD-based storage, data transfer, DNS management, and a static IP
- Amazon Lightsail provides preconfigured virtual private servers (instances) that include everything required to deploy and application or create a database
- Can create Instances and Databases, and configure Static IP, DNS Zone, Load Balancers, storage, and snapshots

Object Storage

- Object-based storage systems manage data as individual objects, rather than as blocks and sectors (block-based) or a file hierarchy (file-based)
- Object-based storage is accessed using a REST API (URL with HTTP methods, e.g. GET, PUT)
- With object storage data is managed as individual objects rather than a file hierarchy (as with a traditional file system)
- Each object includes the data itself, metadata (data about the data), and a globally unique identifier
- Due to its flat file structure, object storage has virtually unlimited scalability and allows the retention of massive amounts of unstructured data

Block Storage

- Data is stored and managed in blocks within sectors and tracks and is controlled by a server-based operating system
- Block storage volumes appear as local disks to the operating system and can be partitioned and formatted
- You can use block storage devices as a boot volume
- Common use cases for block storage are structured information such as file systems, databases, transactional logs, SQL databases and virtual machines (VMs)

File Storage

- File-based storage systems manage data in a file hierarchy
- A file system is mounted via the network to a client computer where it then becomes accessible for reading and writing data
- Protocols used for accessing file systems include NFS or CIFS/SMB

DigitalCloud TRAINING

Amazon Simple Storage Service (S3)

- Amazon S3 is object storage built to store and retrieve any amount of data from anywhere – web sites and mobile apps, corporate applications, and data from IoT sensors or devices
- You can store any type of file in S3
- S3 is designed to deliver 99.999999999% durability
- Typical use cases include:
  - Backup and Storage – Provide data backup and storage services for others
  - Application Hosting – Provide services that deploy, install, and manage web applications
  - Media Hosting – Build a redundant, scalable, and highly available infrastructure that hosts video, photo, or music uploads and downloads
  - Software Delivery – Host your software applications that customers can download
  - Static Website – you can configure a static website to run from an S3 bucket

Amazon S3

- Files are stored in buckets
- Buckets are root level folders
- Files can be anywhere from 0 bytes to 5 TB
- There is unlimited storage available
- S3 is a universal namespace so bucket names must be unique globally
- However, you create your buckets within a REGION
- It is a best practice to create buckets in regions that are physically closest to your users to reduce latency
- Objects consist of:
  - Key (name of the object)
  - Value (data made up of a sequence of bytes)
  - Version ID (used for versioning)
  - Metadata (data about the data that is stored)

Pricing

- Storage
- Requests
- Storage management pricing
- Data transfer pricing
- Transfer acceleration

There are six S3 storage classes:

- S3 Standard (durable, immediately available, frequently accessed)
- S3 Intelligent-Tiering (automatically moves data to the most cost-effective tier)
- S3 Standard-IA (durable, immediately available, infrequently accessed)
- S3 One Zone-IA (lower cost for infrequently accessed data with less resilience)
- S3 Glacier (archived data, retrieval times in minutes or hours)
- S3 Glacier Deep Archive (lowest cost storage class for long term retention)

Amazon Elastic Block Store (EBS)

- Amazon Elastic Block Store (Amazon EBS) provides persistent block storage volumes for use with Amazon EC2 instances in the AWS Cloud
- Each Amazon EBS volume is automatically replicated within its Availability Zone to protect you from component failure, offering high availability and durability
- EBS volume data persists independently of the life of the instance
- EBS volumes do not need to be attached to an instance
- You can attach multiple EBS volumes to an instance
- You cannot attach an EBS volume to multiple instances (use Elastic File Store instead)
- EBS volumes must be in the same AZ as the instances they are attached to
- Termination protection is turned off by default and must be manually enabled
- Root EBS volumes are deleted on termination by default
- Extra non-boot volumes are not deleted on termination by default
- The behaviour can be changed by altering the “DeleteOnTermination” attribute

EBS Snapshots

- Snapshots capture a point-in-time state of an instance
- Snapshots are stored on S3
- Does not provide granular backup (not a replacement for backup software)
- If you make periodic snapshots of a volume, the snapshots are incremental, which means that only the blocks on the device that have changed after your last snapshot are saved in the new snapshot
- Even though snapshots are saved incrementally, the snapshot deletion process is designed so that you need to retain only the most recent snapshot in order to restore the volume
- Snapshots can only be accessed through the EC2 APIs
- EBS volumes are AZ specific but snapshots are region specific

Instance Store Volumes

- Instance store volumes are high performance local disks that are physically attached to the host computer on which an EC2 instance runs
- Instance stores are ephemeral which means the data is lost when powered off (non-persistent)
- Instances stores are ideal for temporary storage of information that changes frequently, such as buffers, caches, or scratch data
- Instance store volume root devices are created from AMI templates stored on S3
- Instance store volumes cannot be detached/reattached

Amazon Elastic File System (EFS)

- EFS is a fully-managed service that makes it easy to set up and scale file storage in the Amazon Cloud
- EFS provides a file system interface and uses the NFSv4.1 protocol
- Good for big data and analytics, media processing workflows, content management, web serving, home directories etc.
- Data is stored across multiple AZ's within a region
- Read after write consistency
- Pay for what you use (no pre-provisioning required)
- Can scale up to petabytes
- EFS is elastic and grows and shrinks as you add and remove data
- Can concurrently connect 1 to 1000s of EC2 instances, from multiple AZs
- A file system can be accessed concurrently from all AZs in the region where it is located
- By default you can create up to 10 file systems per account
- On-premises access can be enabled via Direct Connect or AWS VPN

Amazon EFS

- Instances can be behind an Amazon Elastic Load Balancer
- There are two performance modes:
  - "General Purpose" performance mode is appropriate for most file systems
  - "Max I/O" performance mode is optimized for applications where tens, hundreds, or thousands of EC2 instances are accessing the file system

DigitalCloud TRAINING

VPC Component	What it is
Virtual Private Cloud (VPC)	A logically isolated virtual network in the AWS cloud
Subnet	A segment of a VPC's IP address range where you can place groups of isolated resources
Internet Gateway/Egress-only Internet Gateway	The Amazon VPC side of a connection to the public Internet for IPv4/IPv6
Router	Routers interconnect subnets and direct traffic between Internet gateways, virtual private gateways, NAT gateways, and subnets
Peering Connection	Direct connection between two VPCs
VPC Endpoints	Private connection to public AWS services
NAT Instance	Enables Internet access for EC2 instances in private subnets managed by you)
NAT Gateway	Enables Internet access for EC2 instances in private subnets (managed by AWS)
Virtual Private Gateway	The Amazon VPC side of a Virtual Private Network (VPN) connection
Customer Gateway	Customer side of a VPN connection
AWS Direct Connect	High speed, high bandwidth, *private* network connection from customer to aws
Security Group	Instance-level firewall
Network ACL	Subnet-level firewall

Section 8: Public, Private, and Elastic IP addresses

DigitalCloud TRAINING

Name	Description
Public IP address	**Lost when the instance is stopped** **Used in Public Subnets** **No charge** **Associated with a private IP address on the instance** **Cannot be moved between instances**
Private IP address	**Retained when the instance is stopped** **Used in Public and Private Subnets**
Elastic IP address	**Static Public IP address** **You are charged if not used** **Associated with a private IP address on the instance** **Can be moved between instances and Elastic Network Adapters**

Section 8: NAT Instance vs NAT Gateway

DigitalCloud TRAINING

NAT Instance	NAT Gateway
Managed by you (e.g. software updates)	Managed by AWS
Scale up (instance type) manually and use enhanced networking	Elastic scalability up to 45 Gbps
No high availability – scripted/auto-scaled HA possible using multiple NATs in multiple subnets	Provides automatic high availability within an AZ and can be placed in multiple AZs
Need to assign Security Group	No Security Groups
Can use as a bastion host	Cannot access through SSH
Use an Elastic IP address or a public IP address with a NAT instance	Choose the Elastic IP address to associate with a NAT gateway at creation
Can implement port forwarding through manual customisation	Does not support port forwarding

Virtual Private Cloud

- A virtual private cloud (VPC) is a virtual network dedicated to your AWS account
- Analogous to having your own DC inside AWS
- It is logically isolated from other virtual networks in the AWS Cloud
- Provides complete control over the virtual networking environment including selection of IP ranges, creation of subnets, and configuration of route tables and gateways
- You can launch your AWS resources, such as Amazon EC2 instances, into your VPC
- When you create a VPC, you must specify a range of IPv4 addresses for the VPC in the form of a Classless Inter-Domain Routing (CIDR) block; for example, 10.0.0.0/16
- A VPC spans all the Availability Zones in the region
- You have full control over who has access to the AWS resources inside your VPC
- By default you can create up to 5 VPCs per region
- A default VPC is created in each region with a subnet in each AZ

Security Group	Network ACL
Operates at the instance (interface) level	Operates at the subnet level
Supports allow rules only	Supports allow and deny rules
Stateful	Stateless
Evaluates all rules	Processes rules in order
Applies to an instance only if associated with a group	Automatically applies to all instances in the subnets its associated with

Name	Description
Public IP address	Lost when the instance is stopped Used in Public Subnets No charge Associated with a private IP address on the instance Cannot be moved between instances
Private IP address	Retained when the instance is stopped Used in Public and Private Subnets
Elastic IP address	Static Public IP address You are charged if not used Associated with a private IP address on the instance Can be moved between instances and Elastic Network Adapters

NAT Instance	NAT Gateway
Managed by you (e.g. software updates)	Managed by AWS
Scale up (instance type) manually and use enhanced networking	Elastic scalability up to 45 Gbps
No high availability – scripted/auto-scaled HA possible using multiple NATs in multiple subnets	Provides automatic high availability within an AZ and can be placed in multiple AZs
Need to assign Security Group	No Security Groups
Can use as a bastion host	Cannot access through SSH
Use an Elastic IP address or a public IP address with a NAT instance	Choose the Elastic IP address to associate with a NAT gateway at creation
Can implement port forwarding through manual customisation	Does not support port forwarding

Options for securely connecting to a VPC are:

- AWS managed VPN – fast to setup
- AWS Direct Connect – high bandwidth, low-latency but takes weeks to months to setup
- VPN CloudHub – used for connecting multiple sites to AWS
- Software VPN – use 3rd party software

AWS Direct Connect

- AWS Direct Connect is a network service that provides an alternative to using the Internet to connect a customer's on premise sites to AWS
- Can be used to create a hybrid cloud
- Data is transmitted through a private network connection between AWS and a customer's datacenter or corporate network
- Benefits:
  - Reduce cost when using large volumes of traffic
  - Increase reliability (predictable performance)
  - Increase bandwidth (predictable bandwidth)
  - Decrease latency
- Direct Connect is charged by port hours and data transfer
- Available in 1Gbps and 10Gbps

Section 9: Database Types – Operational vs Analytical

Key differences are ***use cases*** and how the database is ***optimized***

Operational / transactional	Analytical
Online Transaction Processing (OLTP)	Online Analytics Processing (OLAP) – the source data comes from OLTP DBs
Production DBs that process transactions. E.g. adding customer records, checking stock availability (INSERT, UPDATE, DELETE)	Data warehouse. Typically, separated from the customer facing DBs. Data is extracted for decision making
Short transactions and simple queries	Long transactions and complex queries
Relational examples: Amazon RDS, Oracle, IBM DB2, MySQL	Relational examples: Amazon RedShift, Teradata, HP Vertica
Non-relational examples: MongoDB, Cassandra, Neo4j, HBase	Non-relational examples: Amazon EMR, MapReduce

DigitalCloud TRAINING

Data Store	Use Case
Database on EC2	• Need full control over instance and database • Third-party database engine (not available in RDS)
Amazon RDS	• Need traditional relational database • e.g. Oracle, PostgreSQL, Microsoft SQL, MariaDB • Data is well-formed and structured
Amazon DynamoDB	• NoSQL database • In-memory performance • High I/O needs • Dynamic scaling
Amazon RedShift	• Data warehouse for large volumes of aggregated data
Amazon ElastiCache	• Fast temporary storage for small amounts of data

DynamoDB Feature	Benefit
Serverless	Fully managed, fault tolerant, service
Highly available	99.99% availability SLA
NoSQL type of database with Name / Value structure	Flexible schema, good for when data is not well structured or unpredictable
Horizontal scaling	Seamless scalability to any scale with push button scaling or Auto Scaling
DynamoDB Accelerator (DAX)	Fully managed in-memory cache for DynamoDB that increases performance (microsecond latency)
Backup	Point-in-time recovery down to the second in last 35 days; On-demand backup and restore

Section 9: AWS Databases – Relational vs Non-Relational

Key differences are how data are ***managed*** and how data are ***stored***

Relational	Non-Relational
Organized by tables, rows and columns	Varied data storage models
Rigid schema (SQL)	Flexible schema (NoSQL) – data stored in key-value pairs, columns, documents or graphs
Rules enforced within database	Rules can be defined in application code (outside database)
Typically scaled vertically	Scales horizontally
Supports complex queries and joins	Unstructured, simple language that supports any kind of schema
Amazon RDS, Oracle, MySQL, IBM DB2, PostgreSQL	Amazon DynamoDB, MongoDB, Redis, Neo4j

Data Store	Use Case
Database on EC2	• Need full control over instance and database • Third-party database engine (not available in RDS)
Amazon RDS	• Need traditional relational database • e.g. Oracle, PostgreSQL, Microsoft SQL, MariaDB • Data is well-formed and structured
Amazon DynamoDB	• NoSQL database • In-memory performance • High I/O needs • Dynamic scaling
Amazon RedShift	• Data warehouse for large volumes of aggregated data
Amazon ElastiCache	• Fast temporary storage for small amounts of data

DigitalCloud TRAINING

Amazon Relational Database Service (RDS)

- Amazon Relational Database Service (Amazon RDS) is a managed service that makes it easy to set up, operate, and scale a relational database in the cloud
- RDS uses EC2 instances, so you must choose an instance family/type
- Relational databases are known as Structured Query Language (SQL) databases
- RDS is an Online Transaction Processing (OLTP) type of database
- Easy to setup, highly available, fault tolerant, and scalable
- Common use cases include online stores and banking systems
- You can encrypt your Amazon RDS instances and snapshots at rest by enabling the encryption option for your Amazon RDS DB instance
- Encryption uses AWS Key Management Service (KMS)

Amazon RDS

- Amazon RDS supports the following database engines:
  - SQL Server
  - Oracle
  - MySQL Server
  - PostgreSQL
  - Aurora
  - MariaDB
- Scalability: Can only be scaled up by increasing instance size (compute and storage)
- Fault tolerance / disaster recovery with Multi-AZ option
- Automatic failover for Multi-AZ option
- Read replicas option for read heavy workloads

Amazon DynamoDB

- Amazon DynamoDB is a fully managed NoSQL database service that provides fast and predictable performance with seamless scalability
- Push button scaling means that you can scale the DB at any time without incurring downtime
- Data is synchronously replicated across 3 facilities (AZs) in a region
- Amazon DynamoDB global tables provides a fully managed solution for deploying a multi-region, multi-master database
- Amazon DynamoDB Accelerator (DAX) is a fully managed, highly available, in-memory cache for DynamoDB that delivers up to a 10x performance improvement

Amazon RedShift

- Amazon Redshift is a fast, fully managed data warehouse that makes it simple and cost-effective to analyze all your data using standard SQL and existing Business Intelligence (BI) tools
- RedShift is a SQL based data warehouse used for analytics applications
- RedShift is a relational database that is used for Online Analytics Processing (OLAP) use cases
- RedShift is ideal for processing large amounts of data for business intelligence
- RedShift uses Amazon EC2 instances, so you must choose an instance family/type
- RedShift uses columnar data storage
- RedShift always keeps three copies of your data
- RedShift provides continuous/incremental backups

Amazon ElastiCache

- ElastiCache is a web service that makes it easy to deploy and run Memcached or Redis protocol-compliant server nodes in the cloud
- The in-memory caching provided by ElastiCache can be used to significantly improve latency and throughput for many read-heavy application workloads or compute-intensive workloads
- ElastiCache nodes run on Amazon EC2 instances, so you must choose an instance family/type

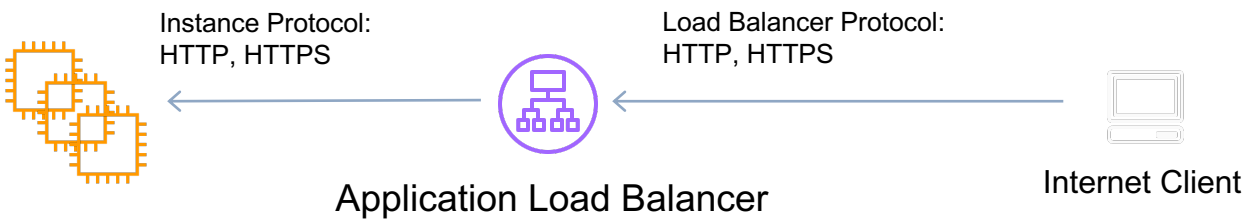
Use Case	Benefit
Web session store	In cases with load-balanced web servers, store web session information in Redis so if a server is lost, the session info is not lost, and another web server can pick it up
Database caching	Use Memcached in front of AWS RDS to cache popular queries to offload work from RDS and return results faster to users
Leaderboards	Use Redis to provide a live leaderboard for millions of users of your mobile app
Streaming data dashboards	Provide a landing spot for streaming sensor data on the factory floor, providing live real-time dashboard displays

Amazon ElastiCache

- There are two types of ElastiCache engine:
  - Memcached – simplest model, can run large nodes with multiple cores/threads, can be scaled in and out, can cache objects such as DBs
  - Redis – complex model, supports encryption, master / slave replication, cross AZ (HA), automatic failover and backup/restore

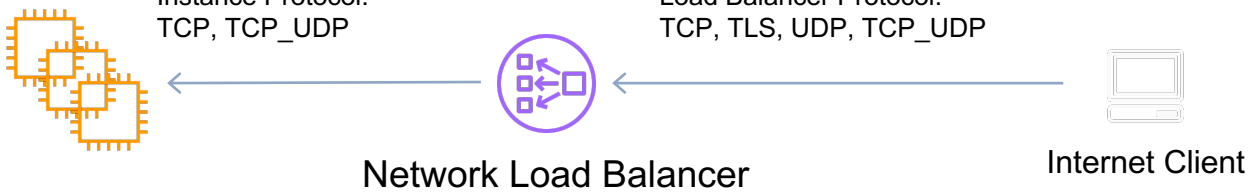
Memcached	Redis
Simple, no-frills	**You need encryption**
You need to elasticity (scale out and in)	You need HIPAA compliance
You need to run multiple CPU cores and threads	Support for clustering
You need to cache objects (e.g. database queries)	You need complex data types
	You need HA (replication
	Pub/Sub capability

Section 10: Elastic Load Balancing (ELB) Types



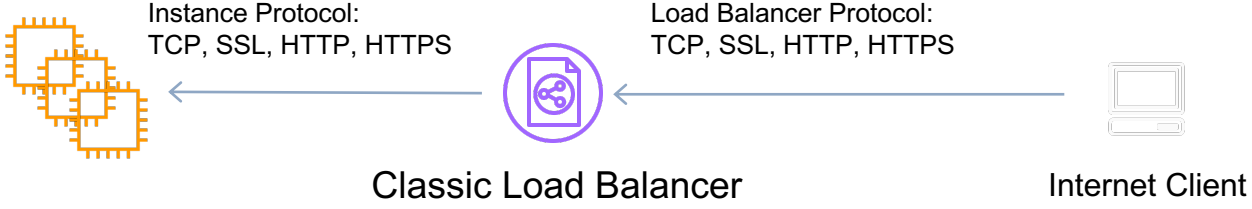
Application Load Balancer

- Operates at the request level
- Routes based on the content of the request (layer 7)
- Supports path-based routing, host-based routing, query string parameter-based routing, and source IP address-based routing
- Supports IP addresses, Lambda Functions and containers as targets



Network Load Balancer

- Operates at the connection level
- Routes connections based on IP protocol data (layer 4)
- Offers ultra high performance, low latency and TLS offloading at scale
- Can have static IP / Elastic IP
- Supports UDP and static IP addresses as targets



Classic Load Balancer

- Old generation; not recommended for new applications
- Performs routing at Layer 4 and Layer 7
- Use for existing applications running in EC2-Classic

Elastic Load Balancing (ELB)

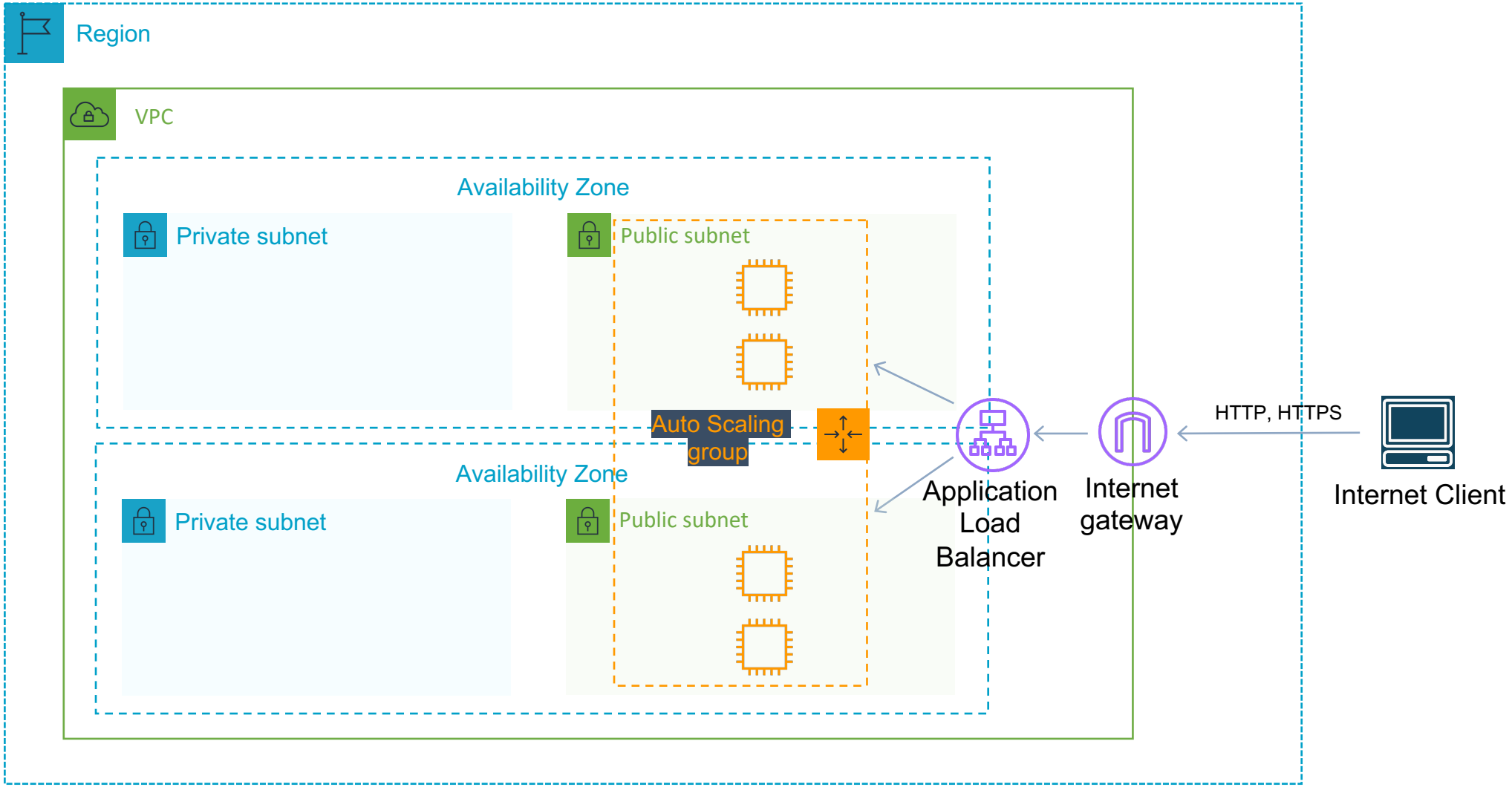
- ELB automatically distributes incoming application traffic across multiple targets, such as Amazon EC2 instances, containers, and IP addresses
- ELB can handle the varying load of your application traffic in a single Availability Zone or across multiple Availability Zones
- ELB features high availability, automatic scaling, and robust security necessary to make your applications fault tolerant
- There are three types of Elastic Load Balancer (ELB) on AWS:
  - Application Load Balancer (ALB) – layer 7 load balancer that routes connections based on the content of the request
  - Network Load Balancer (NLB) – layer 4 load balancer that routes connections based on IP protocol data
  - Classic Load Balancer (CLB) – this is the oldest of the three and provides basic load balancing at both layer 4 and layer 7

DigitalCloud TRAINING

Amazon EC2 Auto Scaling

- Amazon EC2 Auto Scaling provides *horizontal* scaling
- Auto Scaling provides elasticity and scalability
- AWS Auto Scaling automates the process of adding (scaling up) OR removing (scaling down) EC2 instances based on the traffic demand for your application
- Auto Scaling helps to ensure that you have the correct number of EC2 instances available to handle the application load
- You create collections of EC2 instances, called Auto Scaling Group (ASG)
- You can specify the minimum number of instances in each ASG, and AWS Auto Scaling will ensure the group never goes beneath this size
- You can also specify the maximum number of instances in each ASG and the group will never go above this size
- A desired capacity can be configured and AWS Auto Scaling will ensure the group has this number of instances

Section 10: Auto Scaling Group with ALB



Section 11: Route 53 DNS Record Types

DigitalCloud TRAINING

Routing Policy	What it does
Simple	Simple DNS response providing the IP address associated with a name
Failover	If primary is down (based on health checks), routes to secondary destination
Geolocation	Uses geographic location you're in (e.g. Europe) to route you to the closest region
Geoproximity	Routes you to the closest region within a geographic area
Latency	Directs you based on the lowest latency route to resources
Multivalue answer	Returns several IP addresses and functions as a basic load balancer
Weighted	Uses the relative weights assigned to resources to determine which to route to

Amazon Route 53

- Route 53 is the AWS Domain Name Service
- Route 53 is a Global service
- Route 53 performs three main functions:
  - Domain registration – Route 53 allows you to register domain names
  - Domain Name Service (DNS) – Route 53 translates name to IP addresses using a global network of authoritative DNS servers
  - Health checking – Route 53 sends automated requests to your application to verify that it's reachable, available and functional
- DNS failover (automatically change domain endpoint if system fails)
- Integrates with ELB, S3, and CloudFront as endpoints
- Routing policies determine how Route 53 DNS responds to queries

Amazon CloudFront

- CloudFront is a content delivery network (CDN) that allows you to store (cache) your content at “edge locations” located around the world
- This allows customers to access content more quickly and provides security against DDoS attacks
- CloudFront can be used for data, videos, applications, and APIs
- CloudFront benefits:
  - Cache content at Edge Location for fast distribution to customers
  - Built-in Distributed Denial of Service (DDoS) attack protection
  - Integrates with many AWS services (S3, EC2, ELB, Route 53, Lambda)

CloudFront Origins and Distributions:

- An origin is the origin of the files that the CDN will distribute
- Origins can be either an S3 bucket, an EC2 instance, an Elastic Load Balancer, or Route 53 – can also be external (non-AWS)
- To distribute content with CloudFront you need to create a distribution
- There are two types of distribution: Web Distribution and RTMP Distribution

CloudFront uses Edge Locations and Regional Edge Caches:

- An edge location is the location where content is cached (separate to AWS regions/AZs)
- Requests are automatically routed to the nearest edge location
- Regional Edge Caches are located between origin web servers and global edge locations and have a larger cache
- Regional Edge caches aim to get content closer to users

DigitalCloud TRAINING

Amazon CloudWatch

- Amazon CloudWatch is a monitoring service for AWS cloud resources and the applications you run on AWS
- CloudWatch is for performance monitoring (CloudTrail is for auditing)
- Used to collect and track metrics, collect and monitor log files, and set alarms
- CloudWatch is a regional service
- CloudWatch Alarms can be set to react to changes in your resources
- CloudWatch Events generates events when resource states change and delivers them to targets for processing
- CloudWatch Logs collects and centralizes logs from AWS resources
- Any log files generated by your applications
- Gain system-wide visibility into resource utilization
- CloudWatch monitoring includes application performance

DigitalCloud TRAINING

AWS CloudTrail

- AWS CloudTrail is a web service that records activity made on your account and delivers log files to an Amazon S3 bucket
- CloudTrail is for auditing (CloudWatch is for performance monitoring)
- CloudTrail is about logging and saves a history of API calls for your AWS account
- Provides visibility into user activity by recording actions taken on your account
- API history enables security analysis, resource change tracking, and compliance auditing
- Logs API calls made via:
  - AWS Management Console
  - AWS SDKs
  - Command line tools
  - Higher-level AWS services (such as CloudFormation)

DigitalCloud TRAINING

CloudFormation	Elastic Beanstalk
“Template-driven provisioning”	“Web apps made easy"
Deploys infrastructure using code	Deploys applications on EC2 (PaaS)
Can be used to deploy almost any AWS service	Deploys web applications based on Java, .NET, PHP, Node.js, Python, Ruby, Go, and Docker
Uses JSON or YAML template files	Uses ZIP or WAR files (or Git)
CloudFormation can deploy Elastic Beanstalk environments	Elastic Beanstalk cannot deploy using CloudFormation
Similar to Terraform	Similar to Google App Engine

DigitalCloud TRAINING

AWS CloudFormation

- AWS CloudFormation provides a common language for you to describe and provision all the infrastructure resources in your cloud environment
- CloudFormation can be used to provision a broad range of AWS resources
- Think of CloudFormation as deploying infrastructure as code
- Elastic Beanstalk is more focussed on deploying applications on EC2 (PaaS)

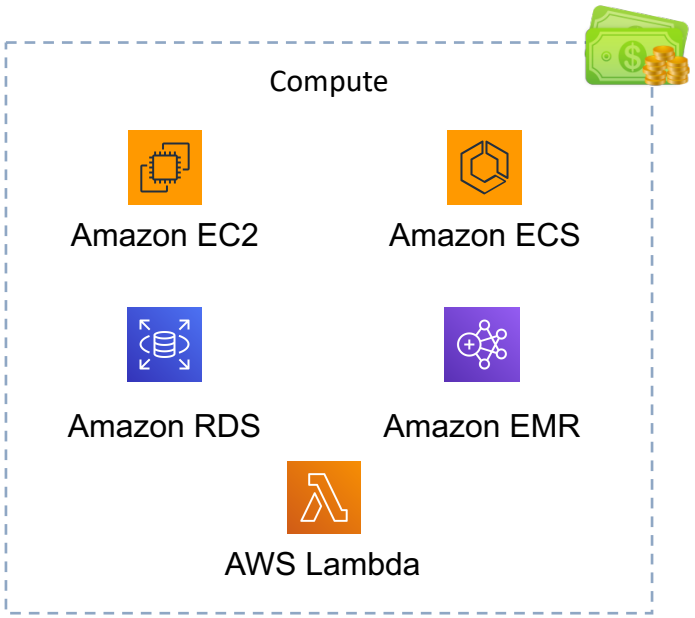
AWS Elastic Beanstalk

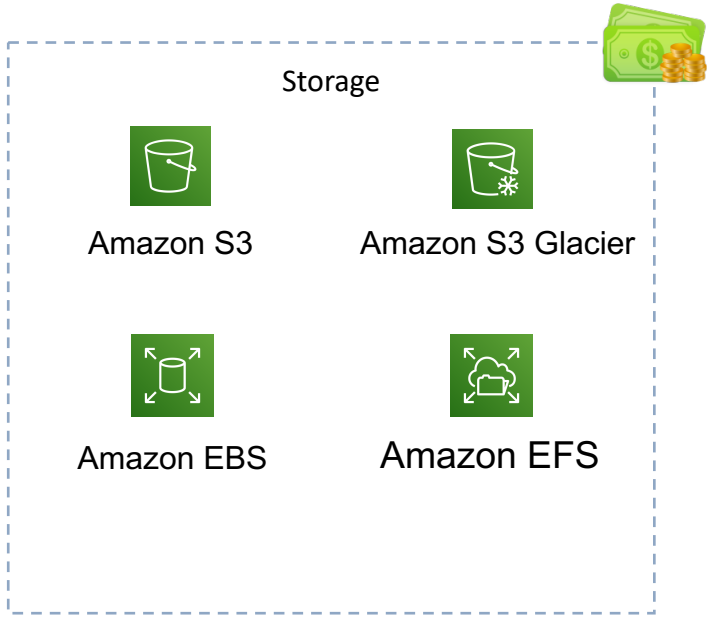
- AWS Elastic Beanstalk can be used to quickly deploy and manage applications in the AWS Cloud
- Developers upload applications and Elastic Beanstalk handles the deployment details of capacity provisioning, load balancing, auto-scaling, and application health monitoring
- Considered a Platform as a Service (PaaS) solution

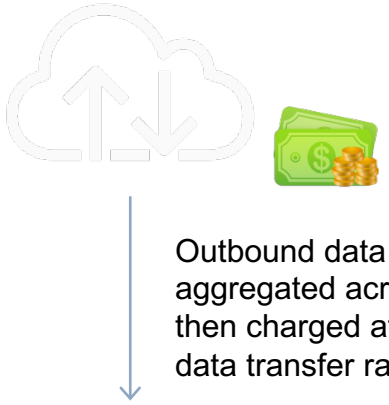
Section 15: Pricing: What you need to know

- You need to know:
  - What you get for free
  - What you get charged for
  - How you are charged – e.g. per GB, per instance hour etc.
  - Special payment options - .e.g. reservations
- You don’t need to know:
  - Exactly how much you get charged

Section 15: Three Fundamentals of AWS Pricing







Outbound data transfer is aggregated across services and then charged at the outbound data transfer rate

Section 15: AWS Pricing Models

- On-demand
  - Used for Compute and Database capacity
  - No long-term commitments or upfront payments
- Dedicated Instances
  - Available for Amazon EC2
  - Hardware is dedicated to a single customer
- Spot Instances
  - Purchase spare capacity with no commitments
  - Great discounts from hourly rates
- Reservations
  - Up to 75% discount in exchange a term commitment

Section 15: Reservations

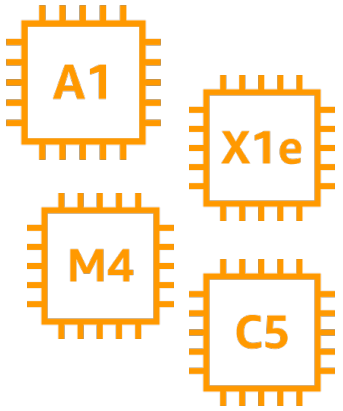
- Options for 1 or 3 year term
- Options to pay:
  - No upfront
  - Partial upfront
  - All upfront
- Available for these services:
  - Amazon EC2 Reserved Instances
  - Amazon DynamoDB Reserved Capacity
  - Amazon ElastiCache Reserved Nodes
  - Amazon Relational Database Service (RDS) Reserved Instances
  - Amazon RedShift Reserved Nodes

Section 15: Free Stuff

- Amazon VPC
- Elastic Beanstalk (but not the resources created)
- CloudFormation (but not the resources created)
- Identity Access Management (IAM)
- Auto Scaling (but not the resources created)
- Consolidated Billing

Amazon EC2 Pricing Components:

- Clock hours of server uptime
- Instance configuration
- Instance type
- Number of instances
- Load balancing
- Detailed monitoring (CloudWatch)
- Auto Scaling (resources created)
- Elastic IP addresses (charged if allocated but not used)
- Operating systems and software packages



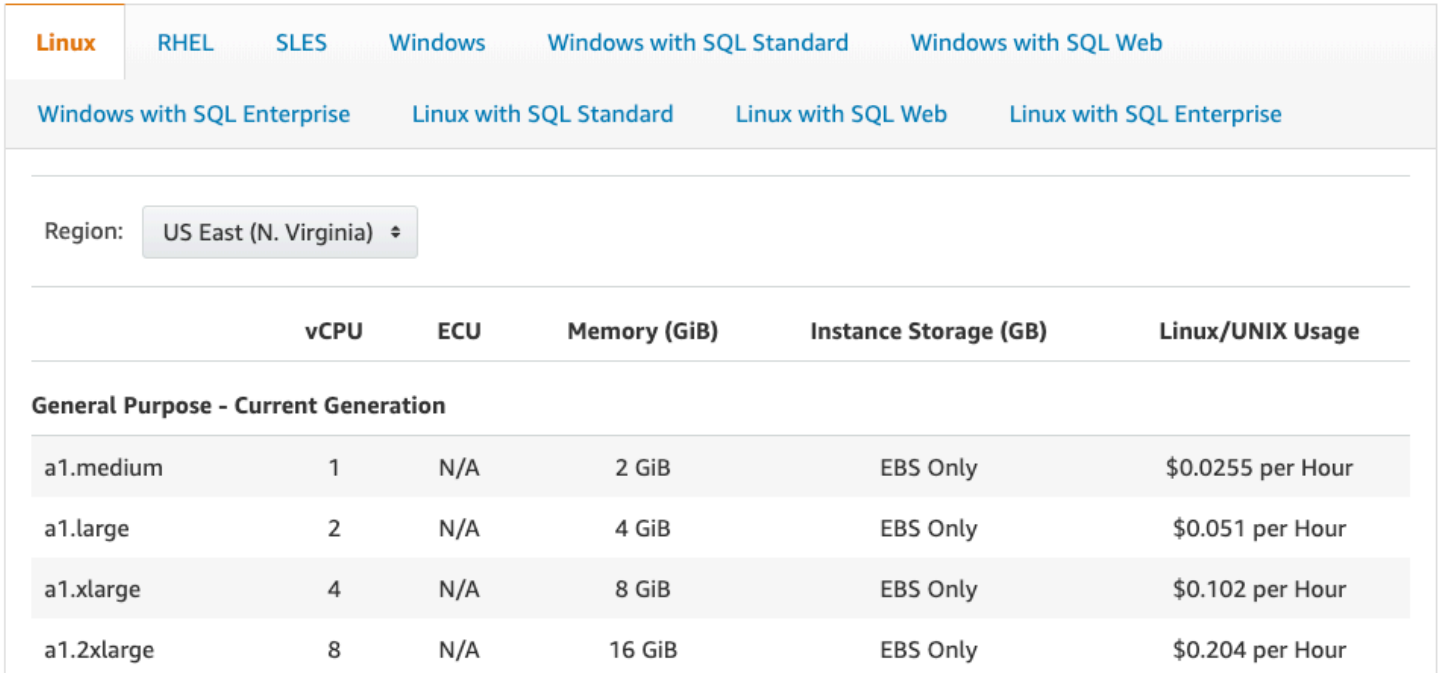
Amazon EC2 Pricing Models

- 5 Pricing models available:
  - On-demand
  - Reserved Instances
  - Spot Instances
  - Dedicated Hosts
  - Savings Plans (new November 2019, not yet on the exam)

DigitalCloud TRAINING

On-Demand

- Means you pay for compute or database capacity with no long-term commitments or upfront payments
- You pay for the compute capacity per hour or per second (per second is Linux only, and applies to On-Demand, Reserved and Spot instances)



Linux | RHEL | SLES | Windows | Windows with SQL Standard | Windows with SQL Web | Windows with SQL Enterprise | Linux with SQL Standard | Linux with SQL Web | Linux with SQL Enterprise

Region: US East (N. Virginia)

	vCPU	ECU	Memory (GiB)	Instance Storage (GB)	Linux/UNIX Usage
General Purpose - Current Generation					
a1.medium	1	N/A	2 GiB	EBS Only	$0.0255 per Hour
a1.large	2	N/A	4 GiB	EBS Only	$0.051 per Hour
a1.xlarge	4	N/A	8 GiB	EBS Only	$0.102 per Hour
a1.2xlarge	8	N/A	16 GiB	EBS Only	$0.204 per Hour

- Recommended for users who prefer low cost and flexibility without upfront payment or long-term commitments
- Good for applications with short-term, spiky, or unpredictable workloads that cannot be interrupted

DigitalCloud TRAINING

Reserved Instances

- Reserved instances provide significant discounts, up to 75% compared to On-Demand pricing, by paying for capacity ahead of time
- Provide a capacity reservation when applied to a specific Availability Zone

STANDARD 3-YEAR TERM					
Payment Option	Upfront	Monthly*	Effective Hourly**	Savings over On-Demand	On-Demand Hourly
No Upfront	$0	$8.03	$0.011	57%	$0.0255
Partial Upfront	$134	$3.72	$0.010	60%	
All Upfront	$252	$0.00	$0.010	62%	

- Good for applications that have predictable usage, that need reserved capacity, and for customers who can commit to a 1 or 3-year term
- Reservation options include no upfront, partial upfront and all upfront

Spot Instances

- Purchase spare computing capacity with no upfront commitment at discounted hourly rates
- Provides up to 90% off the On-Demand price
- Recommended for applications that have flexible start and end times, applications that are only feasible at very low compute prices, and users with urgent computing needs for a lot of additional capacity
- In the old model Spot instances were terminated because of higher competing bids, in the new model this does not happen but instances still may be terminated (with a 2 minute warning) when EC2 needs the capacity back – note: the exam may not be updated to reflect this yet

Dedicated Hosts

- A dedicated host is an EC2 server dedicated to a single customer
- Runs in your VPC
- Good for when you want to leverage existing server-bound software licences such as Windows Server, SQL Server, and SUSE Linux Enterprise Server
- Also good for meeting compliance requirements

Dedicated Instances

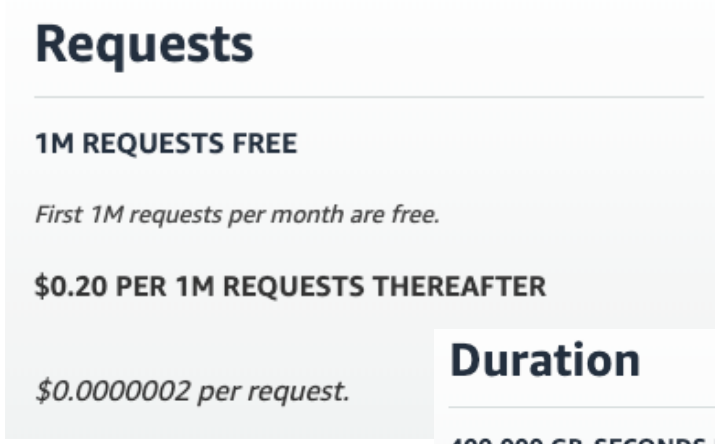
- Dedicated Instances are Amazon EC2 instances that run in a VPC on hardware that's dedicated to a single customer
- Dedicated instances are physically isolated at the host hardware level from instances that belong to other AWS accounts
- Dedicated instances may share hardware with other instances from the same AWS account that are not Dedicated instances

Characteristic	Dedicated Instances	Dedicated Hosts
Enables the use of dedicated physical servers	**X**	**X**
Per instance billing	X	
Per host billing		X
Visibility of sockets, cores, host ID		X
Affinity between a host and instance		X
Targeted instance placement		X
Automatic instance placement	X	X

DigitalCloud TRAINING

AWS Lambda

- Pay only for what you use and charged based on the number of requests for functions and the time it takes to execute the code
- Price is dependent on the amount of memory allocated to the function



Amazon ECS

- With EC2 launch type you pay for the compute instances in your cluster
- With Fargate launch type you pay for the vCPU and memory resources that your containerized application requests

DigitalCloud TRAINING

Amazon S3

- Storage - pricing per class, e.g. Standard or IA
- Storage quantity – data volume stored in your buckets on a per GB basis

S3 Standard Storage	
First 50 TB / Month	$0.023 per GB
Next 450 TB / Month	$0.022 per GB
Over 500 TB / Month	$0.021 per GB

- Number of requests – the number and type of requests, e.g. GET, PUT, POST, LIST, COPY
- Lifecycle transitions requests – moving data between storage classes
- Data transfer – data transferred out of an S3 region is charged
- Transfer Acceleration
- Storage management pricing for some additional features

S3 Inventory††	$0.0025 per million objects listed
S3 Analytics Storage Class Analysis††	$0.10 per million objects monitored per month
S3 Object Tagging	$0.01 per 10,000 tags per month

DigitalCloud TRAINING

Amazon S3 Glacier

- Extremely low cost and you pay only for what you need with no commitments of upfront fees
- Charged for requests and data transferred out of Glacier
- "Amazon Glacier Select" pricing allows queries to run directly on data stored on Glacier without having to retrieve the archive. Priced on amount of data scanned, returned, and number of requests initiated
- Three options for access to archives, listed in the table below:

	Expedited	Standard	Bulk
Data access time	**1-5 minutes**	**3-5 hours**	**5-12 hours**
Data retrievals	$0.03 per GB	$0.01 per GB	$0.0025 per GB
Retrieval requests	On-Demand: $0.01 per request Provisioned: $100 per Provisioned Capacity Unit	$0.050 per 1,000 requests	$0.025 per 1,000 requests

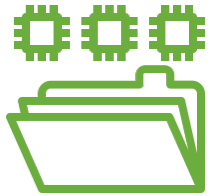
Amazon Elastic Block Store (EBS)

- Volumes – volume storage for all EBS volumes type is charged by the amount of GB provisioned (not used) per month
- Snapshots – based on the amount of space consumed by snapshots in S3. Copying snapshots is charged on the amount of data copied across regions
- Data transfer – inbound data transfer is free, outbound data transfer charges are tiered

Volume

Snapshot

Amazon Elastic File System (EFS)



- Storage classes:
  - Standard
  - Infrequent Access Storage (EFS IA)
- Pay for the amount of storage you use
- Lower cost for EFS IA
- EFS Provisioned throughput

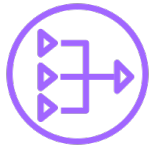
DigitalCloud TRAINING

Amazon Virtual Private Cloud (VPC)

- VPC itself is free, however the following are chargeable:
  - AWS Site-to-Site VPN Connection
  - AWS PrivateLink
  - Amazon VPC Traffic Mirroring
  - NAT Gateways

VPN connection

AWS PrivateLink



NAT gateway

AWS Direct Connect

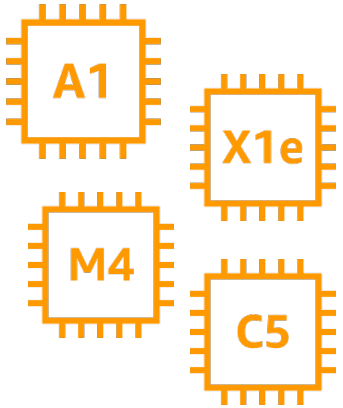
AWS Direct Connect

- Pay for Port Hours
- Outbound data transfer

Capacity	Port-Hour rate (All AWS Direct Connect locations except in Japan)
1G	$0.30/hour
10G	$2.25/hour

Amazon RDS

- Clock hours of server uptime - amount of time the DB instance is running
- Database engine, instance size and memory
- Database purchase type - e.g. On-Demand, Reserved
- Provisioned storage - backup is included up to 100% of the size of the DB
- Requests - the number of input and output requests to the DB
- Deployment type - single AZ or multi-AZ
- Data transfer - inbound is free, outbound data transfer costs are tiered



Amazon DynamoDB

- Provisioned throughput (write)
- Provisioned throughput (read)
- Indexed data storage
- On-demand capacity mode:
  - Pay for the data reads and writes you application makes on the tables
  - No need to specify read and write throughput
- Provisioned capacity mode:
  - Specify the number of reads and writes per second you expect your application to require
  - Can use with Auto Scaling to automatically adjust
- Data transfer
- Global tables
- Reserved Capacity

Amazon CloudFront

- Traffic distribution – data transfer and request pricing, varies across regions, and is based on the edge location from which the content is served
- Requests – the number and type of requests (HTTP or HTTPS) and the geographic region in which they are made
- Data transfer out – quantity of data transferred out of CloudFront edge locations
- There are additional chargeable items such as invalidation requests, field-level encryption requests, and custom SSL certificates

Amazon Route 53

- Monthly charge for managing hosted zones
- DNS queries
- Domain names

AWS Cost Explorer

- The AWS Cost Explorer is a free tool that allows you to view charts of your costs
- You can view cost data for the past 13 months and forecast how much you are likely to spend over the next three months
- Cost Explorer can be used to discover patterns in how much you spend on AWS resources over time and to identify cost problem areas
- Cost Explorer can help you to identify service usage statistics such as:
  - Which services you use the most
  - View metrics for which AZ has the most traffic
  - Which linked account is used the most

AWS Cost Explorer

AWS Simple Monthly Calculator

- The AWS Simple Monthly Calculator helps customers and prospects estimate their monthly AWS bill more efficiently
- With the AWS Simple Monthly Calculator you can add services in different regions
- The calculator includes support for most AWS services and you can include additional costs such as data ingress/egress charges, data storage charges, and retrieval fees
- It is possible to select EC2 dedicated hosts and reserved instances with various pricing models
- Support can also be added

AWS Total Cost of Ownership (TCO) Calculator

- The TCO calculator is a free tool provided by AWS that allows you to estimate the cost savings of using the AWS Cloud vs. using an on-premise data center
- The TCO calculator therefore helps you to reduce Total Cost of Ownership (TCO) by avoiding large capital expenditures on hardware and infrastructure
- The TCO calculator can also provide directional guidance on cost savings
- The TCO calculator works by you inputting cost elements of your current/or estimated on-premises data center, and comparing those cost requirements with how much it would cost on AWS
- Elements can be added/modified as you move through the process to best estimate the cost savings

There are four AWS support plans available:

- Basic – billing and account support only (access to forums only)
- Developer – business hours support via email
- Business – 24×7 email, chat and phone support
- Enterprise – 24×7 email, chat and phone support

- Enterprise support comes with a Technical Account Manager (TAM)
- Developer allows one person to open unlimited cases
- Business and Enterprise allow unlimited contacts to open unlimited cases
- Study the table at: https://aws.amazon.com/premiumsupport/plans/

Section 15: AWS Organizations and Consolidated Billing

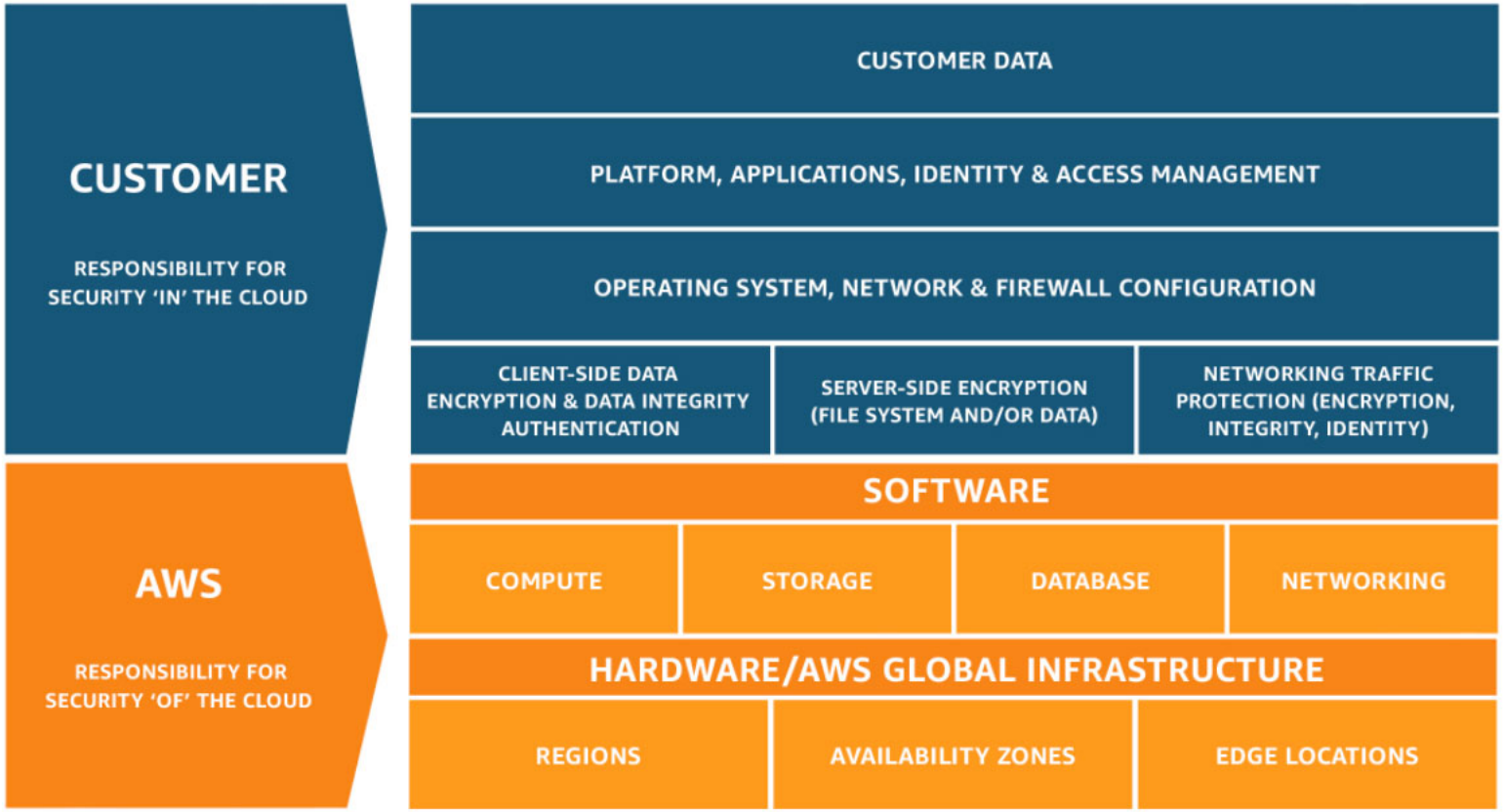
- AWS organizations allows you to consolidate multiple AWS accounts into an organization that you create and centrally manage
- Available in two feature sets:
  - Consolidated Billing
  - All features
- Includes root accounts and organizational units
- Policies are applied to root accounts or OUs
- Consolidated billing includes:
  - Paying Account – independent and cannot access resources of other accounts.
  - Linked Accounts – all linked accounts are independent

- Consolidated billing has the following benefits:
- One bill – You get one bill for multiple accounts
  - Easy tracking – You can track the charges across multiple accounts and download the combined cost and usage data
  - Combined usage – You can combine the usage across all accounts in the organization to share the volume pricing discounts and Reserved Instance discounts. This can result in a lower charge for your project, department, or company than with individual standalone accounts
  - No extra fee – Consolidated billing is offered at no additional cost

Section 15: Resource Groups and Tagging

- Tags are key / value pairs that can be attached to AWS resources
- Tags contain metadata (data about data)
- Tags can sometimes be inherited – e.g. resources created by Auto Scaling, CloudFormation or Elastic Beanstalk
- Resource groups make it easy to group resources using the tags that are assigned to them. You can group resources that share one or more tags
- Resource groups contain general information, such as:
  - Region
  - Name
  - Health Checks
- And also specific information, such as:
  - Public & private IP addresses (for EC2)
  - Port configurations (for ELB)
  - Database engine (for RDS)

Section 16: The Shared Responsibility Model



Section 16: The Shared Responsibility Model

- The AWS shared responsibility model defines what you (as an AWS account holder/user) and AWS are responsible for when it comes to security and compliance
- Security and Compliance is a shared responsibility between AWS and the customer
- AWS are responsible for “Security of the Cloud”
- Customers are responsible for “Security in the Cloud”

- Inherited Controls – Controls which a customer fully inherits from AWS
- Shared Controls – Controls which apply to both the infrastructure layer and customer layers, but in completely separate contexts or perspectives
- Customer Specific – Controls which are solely the responsibility of the customer based on the application they are deploying within AWS services

Section 16: AWS Compliance

- AWS Cloud Compliance enables you to understand the robust controls in place at AWS to maintain security and data protection in the cloud
- As systems are built on top of AWS Cloud infrastructure, compliance responsibilities will be shared
- Compliance programs include:
  - Certifications / attestations
  - Laws, regulations, and privacy
  - Alignments / frameworks

Section 16: AWS Artifact

- AWS Artifact is your go-to, central resource for compliance-related information that matters to you
- It provides on-demand access to AWS' security and compliance reports and select online agreements
- Reports available in AWS Artifact include our Service Organization Control (SOC) reports, Payment Card Industry (PCI) reports, and certifications from accreditation bodies across geographies and compliance verticals that validate the implementation and operating effectiveness of AWS security controls
- Agreements available in AWS Artifact include the Business Associate Addendum (BAA) and the Nondisclosure Agreement (NDA)

AWS Config

- AWS Config is a fully-managed service that provides you with an AWS resource inventory, configuration history, and configuration change notifications to enable security and regulatory compliance
- With AWS Config, you can discover existing and deleted AWS resources, determine your overall compliance against rules, and dive into configuration details of a resource at any point in time. AWS Config enables compliance auditing, security analysis, resource change tracking, and troubleshooting

AWS Service Catalog

- You can use AWS Service Catalog to create and manage catalogs of IT services that you have approved for use on AWS, including virtual machine images, servers, software, and databases to complete multi-tier application architectures
- AWS Service Catalog allows you to centrally manage commonly deployed IT services, and helps you achieve consistent governance to meet your compliance requirements, while enabling users to quickly deploy the approved IT services they need

AWS Key Management Service (KMS)

- AWS Key Management Service gives you centralized control over the encryption keys used to protect your data
- You can create, import, rotate, disable, delete, define usage policies for, and audit the use of encryption keys used to encrypt your data
- AWS Key Management Service is integrated with most other AWS services making it easy to encrypt the data you store in these services with encryption keys you control

AWS CloudHSM

- AWS CloudHSM is a cloud-based hardware security module (HSM) that enables you to easily generate and use your own encryption keys on the AWS Cloud
- With CloudHSM, you can manage your own encryption keys using FIPS 140-2 Level 3 validated HSMs
- CloudHSM offers you the flexibility to integrate with your applications using industry-standard APIs, such as PKCS#11, Java Cryptography Extensions (JCE), and Microsoft CryptoNG (CNG) libraries

Amazon Inspector

- Inspector is an automated security assessment service that helps improve the security and compliance of applications deployed on AWS
- Inspector automatically assesses applications for vulnerabilities or deviations from best practices
- Uses an agent installed on EC2 instances
- Instances must be tagged

AWS Trusted Advisor

- Trusted Advisor is an online resource that helps to reduce cost, increase performance and improve security by optimizing your AWS environment
- Trusted Advisor provides real time guidance to help you provision your resources following best practices
- Advisor will advise you on Cost Optimization, Performance, Security, and Fault Tolerance

Section 16: AWS Personal Health Dashboard

- AWS Personal Health Dashboard provides alerts and remediation guidance when AWS is experiencing events that may impact you
- Personal Health Dashboard gives you a personalized view into the performance and availability of the AWS services underlying your AWS resources
- The dashboard displays relevant and timely information to help you manage events in progress
- Also provides proactive notification to help you plan for scheduled activities
- Alerts are triggered by changes in the health of AWS resources, giving you event visibility, and guidance to help quickly diagnose and resolve issues
- You get a personalized view of the status of the AWS services that power your applications, enabling you to quickly see when AWS is experiencing issues that may impact you

DigitalCloud TRAINING

AWS Web Application Firewall (WAF)

- AWS WAF is a web application firewall
- Protects against common exploits that could compromise application availability, compromise security or consume excessive resources

AWS Shield

- AWS Shield is a managed Distributed Denial of Service (DDoS) protection service
- Safeguards web application running on AWS with always-on detection and automatic inline mitigations
- Helps to minimize application downtime and latency
- Two tiers – Standard and Advanced
- Integrated with Amazon CloudFront

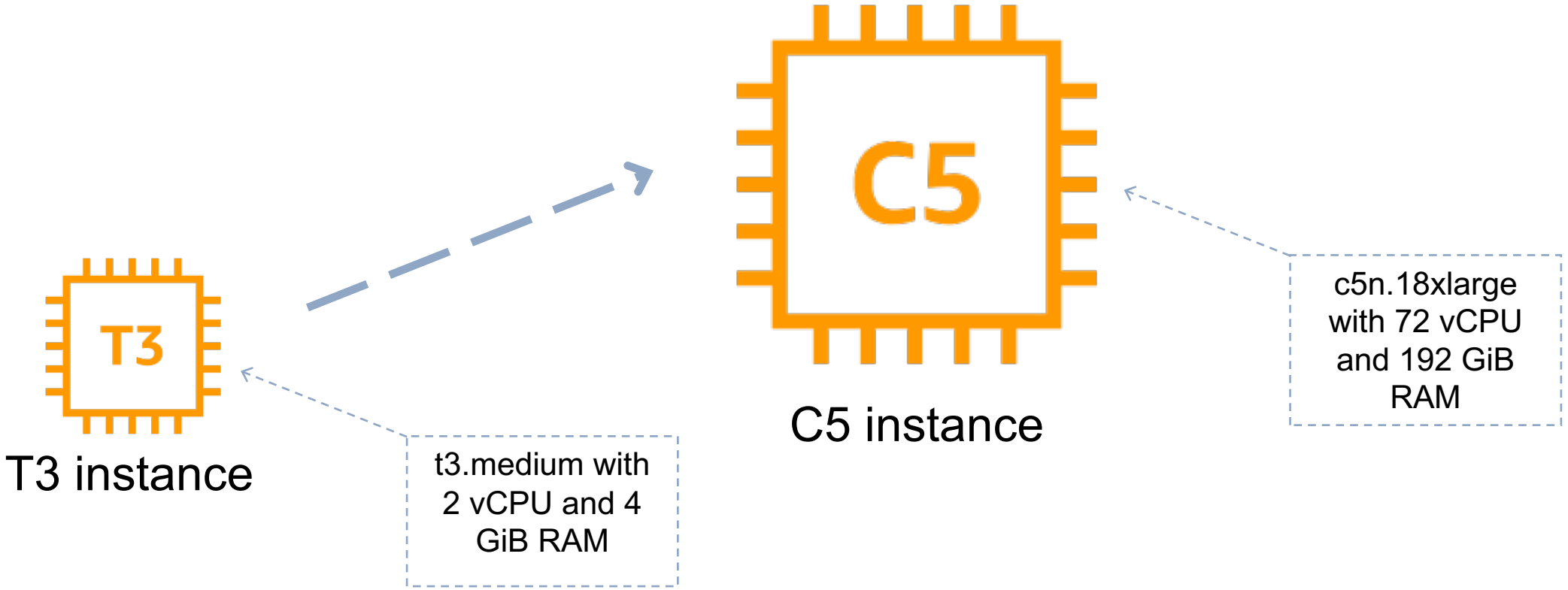
Section 16: Penetration Testing

- Penetration testing is the practice of testing one's own application's security for vulnerabilities by simulating an attack
- AWS allows penetration testing. There is a limited set of resources on which penetration testing can be performed
- You do not need permission to perform penetration testing against the following services:
  - Amazon EC2 instances, NAT Gateways, and Elastic Load Balancers
  - Amazon RDS
  - Amazon CloudFront
  - Amazon Aurora
  - Amazon API Gateways
  - AWS Lambda and Lambda Edge functions
  - Amazon Lightsail resources
  - Amazon Elastic Beanstalk environments

Section 17: The Difference Between Traditional and Cloud Computing Environments

DigitalCloud TRAINING

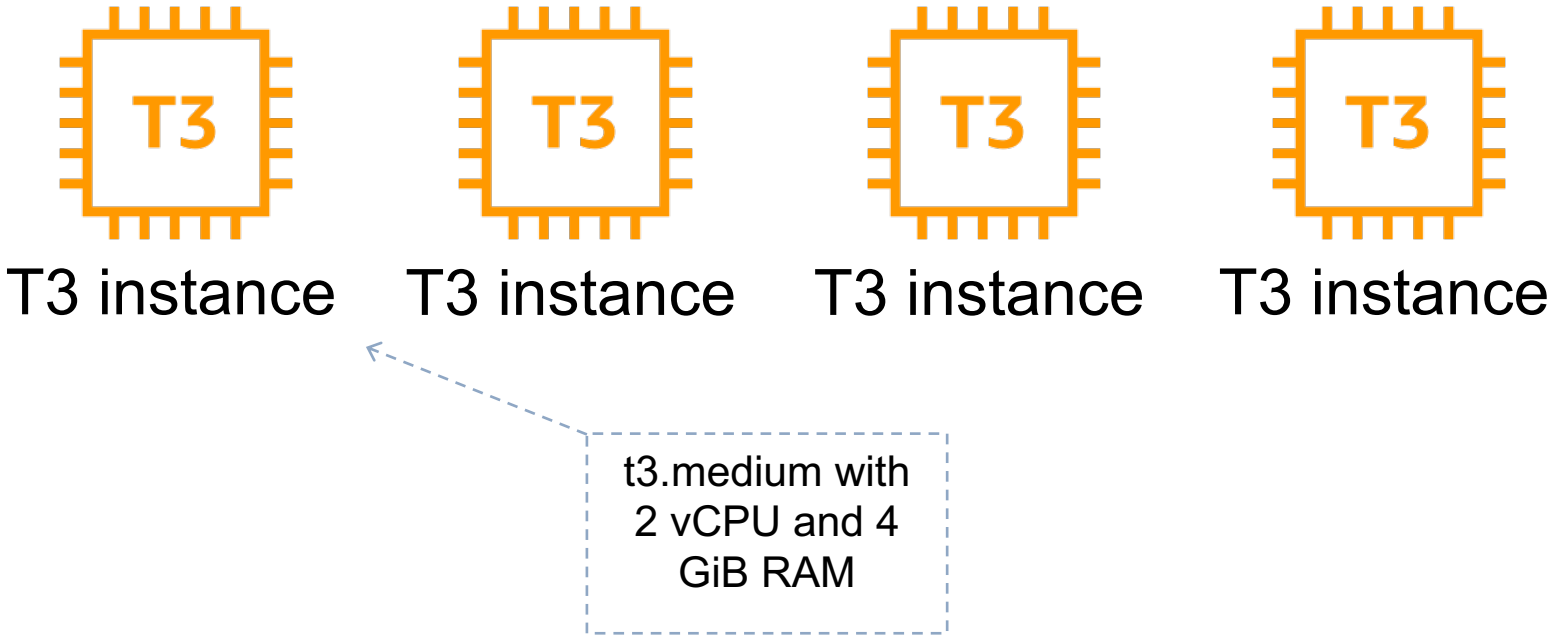
- IT Assets as Provisioned Resources
  - Provision as needed, and scale on-demand
- Global, Available, and Scalable Capacity
  - Deploy globally, easily, cost-effectively, and quickly
- Higher-Level Managed Services
  - Lower operational cost by leveraging managed storage, database, analytics, application and deployment services
- Built-in Security
  - Leverage AWS' significant investment in security, simplify security testing, and use native security and encryption features
- Architecting for Cost
  - Fine grained billing, transparent costs, budgets and alerting tools
- Operating on AWS
  - Tooling, processes and best practices to support operational transition


T3
T3 instance
t3.medium with 2 vCPU and 4 GiB RAM
C5
C5 instance
c5n.18xlarge with 72 vCPU and 192 GiB RAM

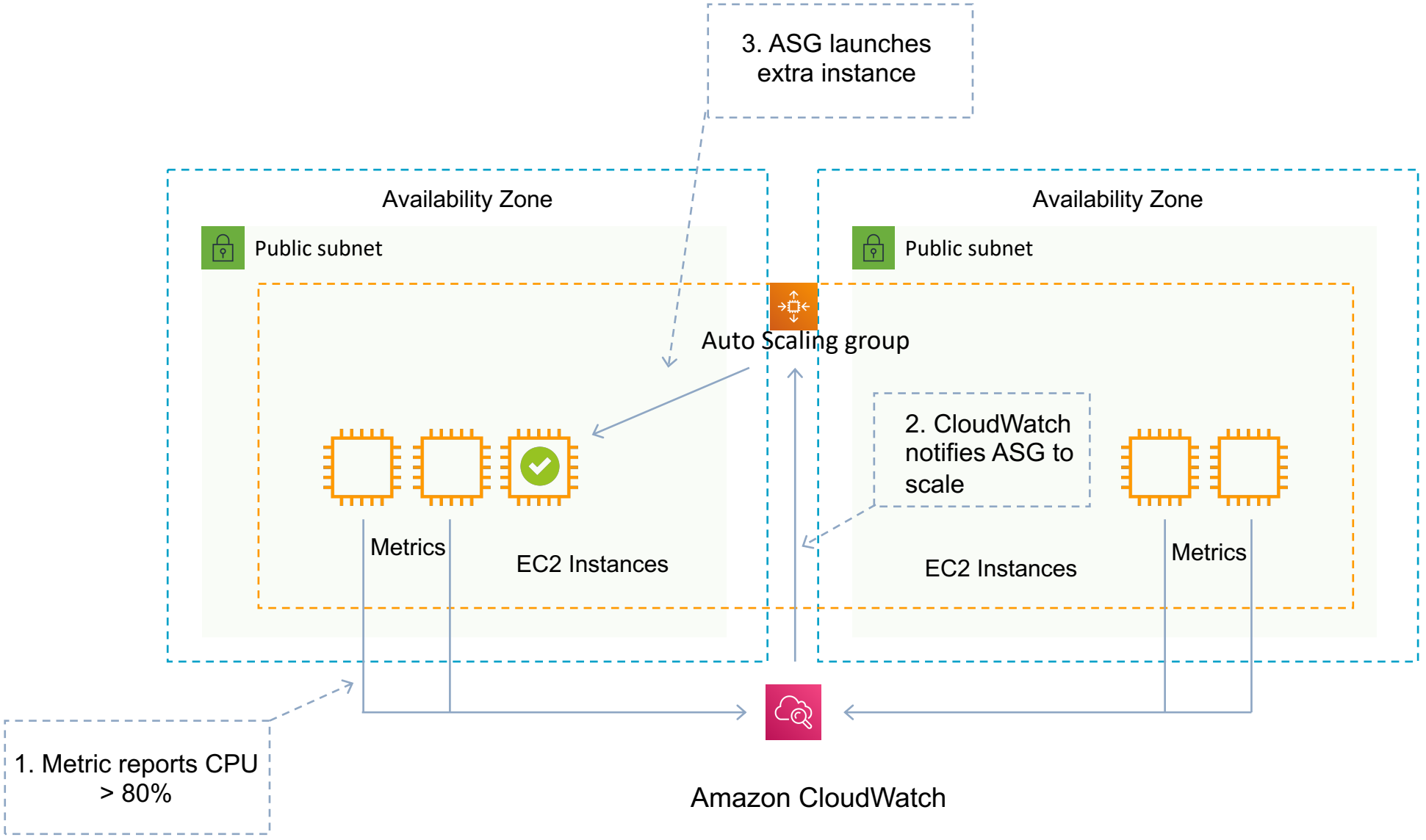
Section 17: Scaling Vertically

- Examples of vertical scaling are:
  - Amazon EC2 instances
  - Amazon RDS Database instances

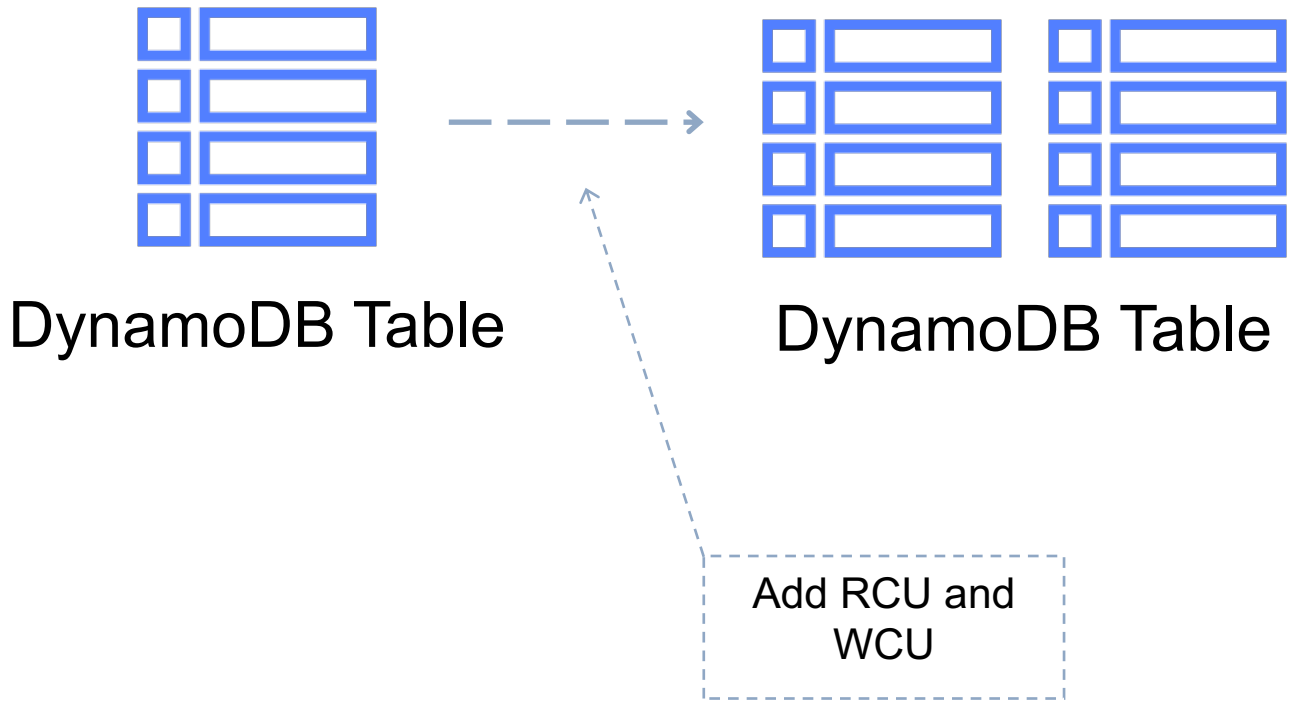
- Limitations of scaling vertically:
  - Often requires manual intervention (though can be scripted/automated)
  - Typically requires downtime
  - Can reach a limit of scalability


T3
T3
T3
T3
T3 instance
T3 instance
T3 instance
T3 instance
t3.medium with
2 vCPU and 4
GiB RAM

Section 17: Example of Scaling Horizontally with EC2



Section 17: Example of Horizontal Scaling with DynamoDB



Section 17: Scaling Horizontally

- Examples of horizontal scaling are:
  - Amazon EC2 Auto Scaling
  - Amazon DynamoDB

- Benefits of scaling horizontally:
  - Seamless scaling, without downtime
  - Can scale almost limitlessly (in some cases)

DigitalCloud TRAINING

Architecting for the Cloud: AWS Best Practices

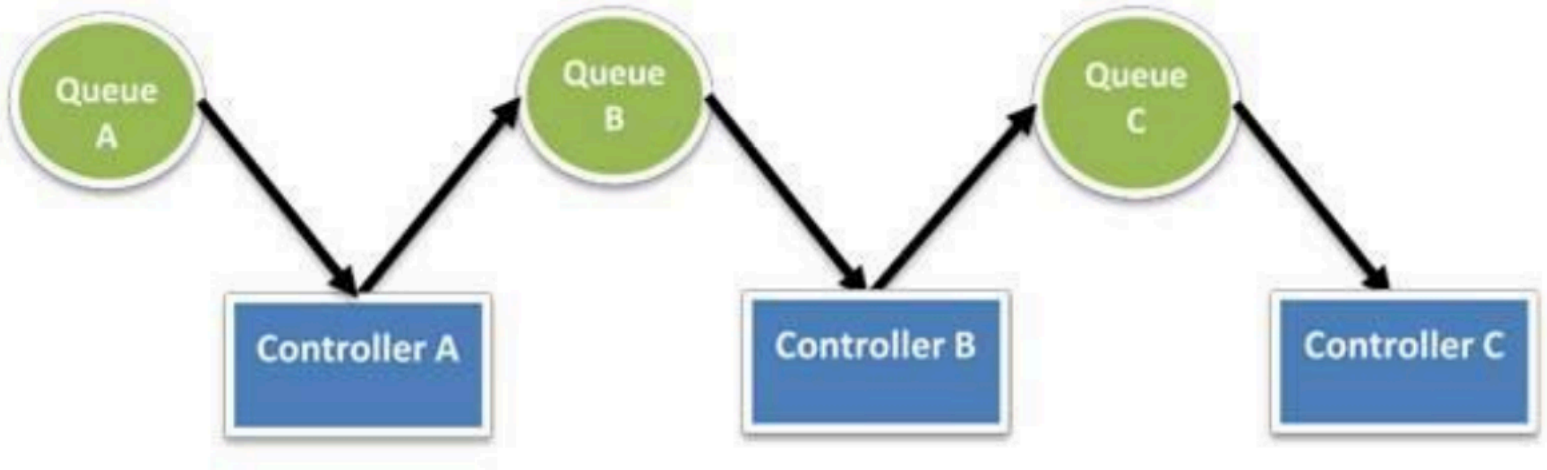
- With traditional infrastructure you work with fixed resources due to upfront costs and lead time
- With the cloud you can take advantage of the dynamically provisioned resources
- When designing for AWS, you can take advantage of the dynamically provisioned nature of cloud computing
- You can think of servers and other components as temporary resources
- You can launch as many as you need, and use them only for as long as you need them

Architecting for the Cloud: AWS Best Practices

- Use automation tools on AWS to improve system stability and organizational efficiency
- Examples are:
  - Serverless Management and Deployment
  - Infrastructure Management and Deployment
  - Alarms and Events

Architecting for the Cloud: AWS Best Practices

- Design IT systems to reduce interdependencies
- A change or a failure in one component should not cascade to other components
- Examples are:
  - Service Discovery – components should be able to “find” each other
  - Asynchronous Integration – introduce a layer between application components to store messages



Example of asynchronous integration with Amazon SQS

DigitalCloud TRAINING

Architecting for the Cloud: AWS Best Practices

- This best practices advises customers to leverage more than just Amazon EC2
- Try to use the breadth of services available on AWS
- For example:
  - Amazon CloudFront for content delivery
  - ELB for load balancing
  - Amazon DynamoDB for NoSQL databases
  - Amazon CloudSearch for search workloads
  - Amazon Elastic Transcoder for video encoding

DigitalCloud TRAINING

Architecting for the Cloud: AWS Best Practices

- Use the right database technology for each workload
- Consider performance requirements, scalability, durability, functionality etc.
- Choose between Relational, NoSQL, Data Warehouse, Search, and Graph

Architecting for the Cloud: AWS Best Practices

- Design for failure!
- Introduce redundancy – ensure if a component / resource fails, another can take over
- Detect failure – use health checks and alarms and try to automate detection and reaction
- Durable Data Storage – ensure your data aligns with your Recovery Point Objective (RPO) and Recovery Time Objective (RTO)
- Automated Multi-Data Center Resilience – be resilient in the face of a major disaster

DigitalCloud TRAINING

Architecting for the Cloud: AWS Best Practices

- Right Sizing – use the best instance sizes and number of instances for cost efficiency
- Elasticity – horizontally scale as needed with changing demand
- Take Advantage of the Variety of Purchasing Options – use Reserved Instances and Spot Instances

DigitalCloud TRAINING

Architecting for the Cloud: AWS Best Practices

- Caching can be used to improve performance and cost efficiency
- Methods of caching include:
  - Application Data Caching – examples are Amazon ElastiCache and DynamoDB DAX
  - Edge Caching – key example is Amazon CloudFront

Architecting for the Cloud: AWS Best Practices

- Use AWS Features for Defense in Depth
- Share Security Responsibility with AWS
- Reduce Privileged Access
- Security as Code – for instance, use CloudFormation to repeatably build secure environments
- Real-Time Auditing – tools such as Trusted Advisor, AWS Config, and Amazon Inspector

Section 17: The Five Pillars of Operational Excelle

DigitalCloud TRAINING

1) Operational Excellence

- The operational excellence pillar includes the ability to run and monitor systems to deliver business value and to continually improve supporting processes and procedures

2) Security

- The security pillar includes the ability to protect information, systems, and assets while delivering business value through risk assessments and mitigation strategies

3) Reliability

- The reliability pillar includes the ability of a system to recover from infrastructure or service disruptions, dynamically acquire computing resources to meet demand, and mitigate disruptions such as misconfigurations or transient network issues

4) Performance Efficiency

- The performance efficiency pillar includes the ability to use computing resources efficiently to meet system requirements and to maintain that efficiency as demand changes and technologies evolve

5) Cost Optimization

- The cost optimization pillar includes the ability to avoid or eliminate unneeded cost or suboptimal resources