AWS Cloud Practitioner Study Guide

Exam CLF-001

Revised: April 2023

Chapter 1

Fundamental Concepts

1.1 Benefits of Cloud

Six Advantages of Cloud Computing

1. **Trade fixed expense for variable expense** - Pay-as-you go pricing in the cloud allows you to pay only for the resources you use and eliminates the need for large up-front infrastructure investments. Trade capital expense (CAPEX) for operational expense (OPEX).
2. **Benefit from economies of scale** - You can achieve a lower cost with cloud resources than you could do on your own. Because AWS is managing resources for thousands of users you benefit from economies of scale.
3. **Stop guessing capacity** - Eliminate the need to guess about infrastructure capacity needs. Systems deployed into AWS can scale up or down as needed with minimal notice time.
4. **Increase speed and agility** - In the cloud new resources are available quickly with a few clicks. This improves organizational agility by getting resources to developers much more quickly.
5. **Stop spending money running and maintaining data centers** - With cloud computing there is no data center to maintain and operate. This results in cost savings and allows you to focus on the business and not infrastructure.
6. **Go global in minutes** - With cloud computing you have access to compute resources worldwide with just a few minutes.

Additional advantages of cloud architectures include:

- **Elasticity** - Ability to acquire resources when needed and release them when no longer needed.
- **Reliability** - A solution's ability to provide its intended functionality when needed.
- **Agility** - Lowers cost of trying new things; less time maintaining infrastructure; access to emerging technologies; reduced risk around security and compliance.

1.2 Cloud Computing Models

- **Infrastructure as a Service (IaaS)** - This model gives the greatest amount of flexibility and control. IaaS services are very similar to working with your own physical infrastructure, but you are working with virtualized infrastructure components without access to the underlying hardware.
- **Platform as a Service (PaaS)** - This model is designed to remove the burden of managing infrastructure resources such as compute, storage, etc. This service offers a platform onto which you deploy your software

applications.

- **Software as a Service (SaaS)** - With this model applications are fully hosted and managed by the provider. These services take away the need to manage any infrastructure or software.

1.3 Cloud Deployment Models

- **Public Cloud** - Deployment onto a public cloud provider like AWS, Azure, or GCP.
- **Private Cloud (on-premises)** - Deployment onto a cloud-like platform in your own data center.
- **Hybrid** - Applications deployed in an environment that is a mixture of both public and private cloud components.

1.4 AWS Global Infrastructure

- **AWS Region** - A cluster of data centers in a specific geographic location working together and providing cloud services.
- **Availability Zone** - One or more data centers within a specific AWS Region with redundant power, networking, and connectivity. Each region will have multiple availability zones.
- **AWS Edge Locations** - Data center locations used as a part of a global content delivery network (CDN). These locations only apply to global services such as CloudFront (CDN) and Route 53 (DNS), Web Application Firewall, and AWS Shield. There are over 200 edge locations in 47 different countries.

1.5 Shared Responsibility Model

The **customer** is responsible for security of services running in the cloud. This includes OS level security patches on EC2, software configuration / patching for custom or third-party software.

AWS is responsible for security of the cloud. This includes the AWS network and server infrastructure, and underlying operating systems for managed services that do not provide OS level access.

Chapter 2

Cost & Support Options

2.1 Cloud Economics

Traditional data centers have a very large capital (up-front) expense to get started, plus operational costs for the lifetime of the data center. Additional costs are incurred anytime increased capacity is needed. Cloud on the other hand does not have any up-front expenses. Operational expenses will scale based on your demand/usage (i.e. pay as you go). Traditional data centers also have the potential for unused capacity or unmet demand, but with cloud computing you can scale compute resources to fit your need.

To help manage and predict costs of operating in the cloud AWS provides several tools:

- **Cost Explorer** - Cost Explorer is a user interface for exploring your costs on AWS. This service allows for cost breakdowns based on service or cost tag. Provides predictions for the next three months of cost based on previous usage, and recommendations for cost optimization.
- **Budgets** - Allows for planning and tracking usage across AWS services. Supports setting alerts based on different spend criteria.
- **TCO Calculator** - Enables determination of costs for leveraging cloud infrastructure.
- **Simple Monthly Calculator** - Calculate costs of running specific AWS services/infrastructure.
- **Organizations** - Allows for managing multiple accounts under a single master account with consolidated billing and clean separation of costs between organizations. Also, provides for centralized logging and security standards.
- **Resource Tags** - Metadata that can be attached to specific resources. These are stored as key/value pairs and can be used to identify department, environment, project, etc.

2.2 Consulting Services

- **AWS Professional Services** - A global team of experts that can help you with your business objectives in the cloud by working with your team and your chosen member of the AWS Partner Network.
- **AWS Partner Network (APN)** - A global community of partners that provide consulting services to help customers migrate to and leverage AWS cloud capabilities.

2.3 AWS Support Options

AWS offers different paid support plans based on communication methods, response time, cost, and type of guidance offered.

Table 2.1: Comparison of AWS Support Plans

	Basic	**Developer**	**Business**	**Enterprise**
Starting Cost	Included	$29/mo	$100/mo	$15,000/mo
Target Audience		Individual Devs		
Documentation, forums, etc.	24x7	24x7	24x7	24x7
Trusted Advisor (7 Checks)	Yes	Yes		
Trusted Advisor (All Checks)			Yes	Yes
Personal Health Dashboard	Yes	Yes	Yes	Yes
Business hours email support		Yes	Yes	Yes
Designated Contacts		1	Unlimited	Unlimited
Dedicated Account Manager				Yes
General Guidance Response Time		24 business hr	24 business hr	24 business hr
System Impaired Response Time		12 business hr	12 business hr	12 business hr
Prod. Sys Impaired Response Time			4 hr	4 hr
Prod. Sys Down Response Time			1 hr	1 hr
Business Sys Down Response Time				15 min

Chapter 3

Services Overview

3.1 Compute Services

- **EC2** - Elastic Compute Cloud provides access to on-demand compute resources. EC2 is basically a virtual machine running in the cloud on AWS infrastructure.
- **Elastic Beanstalk** - Automates the deployment and scaling of workloads on EC2 (PaaS). You only pay for other services used and not for elastic beanstalk itself. Works for specific application platforms (Java, .NET, Node, Docker, etc).
- **AWS Lambda** - Platform for running serverless applications (cloud functions). You pay based on function execution time and memory.
- **Elastic Container Service (ECS)** - Run docker containers on AWS. Customer must provision and maintain the required EC2 instances.
- **Fargate** - Run docker containers on AWS without needing to manage infrastructure. Serverless approach to containers.
- **Elastic Container Registry (ECR)** - Private docker registry on AWS for storing images for ECS/Fargate.
- **AWS Batch** - Run batch processing jobs at any scale. Automatically launches EC2 instances with the required compute and memory resources. Docker images running on ECS.
- **Amazon Lightsail** - Low cost and predictable pricing. Simpler alternative to other services for basic use-cases. Virtual servers, storage, databases, and networking.

3.2 Content and Network Delivery Services

- **Amazon Virtual Private Cloud (VPC)** - A VPC is a service that defines a virtual environment for your application(s) that is an isolated section of AWS' cloud environment. A VPC is a "container" in which you run all of your services. It is a logically separated part of AWS for your use and provides virtual networking configurable for your application.
- **AWS Direct Connect** - Allows for establishing a connection from AWS directly to your data center.
- **AWS PrivateLink** - Allows for establishing secure connections between two VPCs and AWS services without exposing the services to the internet.
- **Amazon Route 53** - Global service (no regions) for providing DNS. Utilizes AWS Edge Locations.
- **Elastic Load Balancing** - Cloud based load balancers to distribute traffic across multiple EC2, ECS, or Lambda targets. Multiple types of load balancers available: application (ALB), network (NLB), or classic.

- **CloudFront** - Global content delivery network (CDN) utilizing AWS edge locations. Can be used for both static and dynamic content. Includes security features such as DDoS protection and web application firewall (WAF). Utilizes AWS Edge Locations.
- **Amazon API Gateway** - Fully managed API service. Integrates with multiple AWS services and provides monitoring and metrics on API calls.
- **AWS Global Accelerator** - Routes via AWS global edge locations. Once a request reaches an edge location it is routed within AWS network instead of internet. Improves performance and has superior fault tolerance.
- **AWS Local Zones** - Allows you to run AWS services for your application close to large population centers to reduce latency to your end users.

3.3 File Storage Services

- **Amazon Simple Storage Service (S3)** - Fully managed file storage service that stores files as objects in buckets. Storage across multiple AZs with multiple storage classes and lifecycle policies.
- **Elastic Block Store (EBS)** - Persistent block storage for use on EC2 instances. Allows for redundancy within AZ and can scale to support petabytes of data.
- **Elastic File System (EFS)** - Fully managed NFS file system designed for linux workloads. Supports petabyte scale and storage across multiple AZs.
- **Amazon FSx** - Fully managed Windows file system with SMB/NTFS support and AD integration.
- **AWS Snow Ball** - Physical device shipped to your location to move petabyte scale information into S3.
- **AWS Snow Mobile** - Shipping container brought to your location for moving exabyte scale information into S3.

3.4 Database Services

- **Amazon Relational Database Service (RDS)** - Fully managed relational database service supporting common database engines such as MySQL, PostgreSQL, Oracle, etc.
- **Amazon Aurora** - Amazon Aurora is an AWS specific database engine providing compatability with MySQL and PostgreSQL. This is a cloud native database solution that supports automatic scaling of storage in increments of 10GB. Can store up to 64TB.
- **Amazon Database Migration Service (DMS)** - Tooling for one time or continual data migration into Amazon RDS from many different commercial database systems.
- **Amazon Dynamo DB** - Fully managed no SQL DB. Support key-value storage or document storage. Has extremely low latency and automatic scaling. Can handle up to 20 millions requests per second. Can store 100s of terabytes.
- **Amazon ElastiCache** - Fully managed in memory datastore with on demand scaling and replication. Used for caching and session storage. Compatible with redis and memcached.
- **Amazon Red Shift** - Data warehouse service supporting high performance and petabyte scale. Uses column oriented storage.
- **Amazon Neptune** - Fully managed graph oriented database.
- **Amazon QLDB** - Quantum Ledger Database is a managed service that provides transparent, immutable, and cryptographically verifiable transaction log (journal).
- **Amazon Managed Blockchain** - Fully managed service for building blockchain application using the open-source frameworks Hyperledger Fabric and Ethereum.

3.5 Integration Services

- **Simple Notification Service (SNS)** - Publish / subscribe messaging service supporting topics. Supports notifications via SMS, email, or push.
- **Simple Queue Service (SQS)** - Message queueing service that supports up to 256k payload. Message can be stored up to 14 days. Supports standard (order not guaranteed) or FIFO (order guaranteed) queues.
- **AWS Step Functions** - Serverless tool for orchestration of workflows. Amazon states language (DSL) for defining workflows.

3.6 Management and Governance Services

- **AWS Cloud Trail** - Logging and monitoring of interactions across all AWS services into S3 buckets. Used for compliance, forensic analysis, operational analysis, and troubleshooting.
- **AWS Cloud Watch** - Logging service for your application and AWS services. Metrics, alarms, and logs for infrastructure.
- **AWS Config** - Monitors and records configuration and history of infrastructure.
- **AWS Systems Manager** - Set of tools for managing infrastructure. Secure way to access server using AWS credentials.
- **AWS Cloud Formation** - Automates provisioning of cloud infrastructure by building templates in YAML or JSON.
- **AWS OpsWorks** - Managed instances of chef and puppet.
- **AWS Organizations** - Provides consolidated billing, logging, security, and account management for multiple accounts under a single master account.
- **AWS Control Tower** - Centralize user access and management across multiple accounts.

3.7 Identity and User Management

- **AWS Identity and Access Management (IAM)** - Service for defining users and controlling access to AWS services. Performs authentication and authorization management (including multi-factor authentication).
- **Amazon Cognito** - Manage authentication / authorization for your custom applications and provides a fully managed user directory service. Enabled controlled access to AWS resources (e.g. S3 bucket) and works with enterprise IdPs (e.g. AD, SAML).

AWS Identity and Access Management (IAM)

- **Users** - Represent the individuals using your AWS account. Account should be created for each user rather than using the root account.
- **Groups** - A collection of users that allow for assigning permissions to multiple users at once.
- **Permissions** -
- **Policies** - Policies define permissions for an action regardless of the method you use to perform the operation.
- **Roles** - An IAM role is an entity that you can create within an AWS account and associate specific permissions. Roles are similar to an IAM user, but it is not associated with a specific person. Instead it can be accessible to anyone that needs it.

3.8 Security Services

- **AWS Key Management Service (KMS)** - Service for managing encryption keys used within AWS.
- **AWS Certificate Manager (ACM)** - Easily provision, manage, deploy, and renew SSL/TLS certificates.
- **AWS Web Application Firewall (WAF)** - Protects web applications from common exploits at the HTTP level (layer 7). Extra cost option to deploy on an application load balancer, API Gateway or on cloud front. Utilizes AWS Edge Locations.
- **AWS Shield** - Free service that is automatically activated for all customer. Protection from some layer 3/4 attacks such as SYN/UDP floods and reflection attacks. Utilizes AWS Edge Locations.
- **AWS Config** - Audit current configuration and changes within the AWS environment.
- **Amazon Macie** - Data security and privacy service that uses machine learning and pattern matching to protect your sensitive data in the cloud (e.g. detection of PII data).
- **Amazon Guard Duty** - Intelligent threat discovery using machine learning. Monitors DNS logs, CloudTrail Events, VPC logs, and Kubernetes logs and can generate CloudWatch events to notify of findings.
- **Amazon Inspector** - Automated security assessments for EC2 instances and containers pushed to Amazon ECR.

Controlling network traffic within a VPC.

AWS provides two services for controlling traffic between services within a VPC and between the VPC and the outside world: Security Groups and Network ACLs. Instances can be secured using only security groups, but network ACLs can also be added as an additional layer of protection.

- **Security Group** - Allow for defining of inbound and outbound traffic at the resource level (e.g. for an EC2 instance). Resource instances can belong to one or more security groups. If a security group is not specified then the default security group for the VPC is used.
- **Network ACL** - Define rules at the subnet level for inbound and outbound traffic. This is much like a more traditional firewall.

Security Group	Network ACL
Operates at the instance level.	Operates at the subnet level.
Only applies to instances it is specifically associated with.	Applies to all instances in the associated subnet.
Only supports defining ALLOW rules.	Supports defining rules as either ALLOW or DENY.
Evaluates all rules before deciding if traffic is allowed.	Evaluates rules in order starting with the lowest numbered rule.
Stateful: Return traffic is always allowed regardless of other rules.	Stateless: Return traffic must be explicitly allowed by the rules.

3.9 Data Processing Services

- **AWS Glue** - Serverless extract, transform, and load (ETL) service for RDS, DynamoDB, RedShift, S3.
- **Amazon EMR** - Elastic map reduce service for big data processing (includes Spark, Flink, Hive, HBase, Hudi, and Presto support).
- **AWS Data Pipeline** - Managed extract, transform, and load (ETL) service with data workflows.
- **Amazon Athena** - Managed service for querying large scale data in S3 using standard SQL style queries.
- **Amazon Quick Sight** - Managed business intelligence service and dynamic dashboards.

- **Amazon Cloud Search** - Managed service that enables developers to build search capabilities into custom applications.
- **Amazon Rekognition** - Computer vision image/video recognition used to identify object, actions, or perform facial recognition.
- **Amazon Translate** - Text translation service for 54 different languages.
- **Amazon Transcribe** - Speech recognition / transcription for 31 different languages.

3.10 Developer Tools

- **AWS CodeCommit** - Private Git repository hosting on AWS cloud.
- **AWS CodeBuild** - CI tool that compiles code, runs test, and produces deployable software packages.
- **AWS CodePipeline** - Continuous delivery tool for building build/deployment pipelines.

3.11 AI & Machine Learning

- **Amazon SageMaker** - Fully managed infrastructure, tools, and workflows for building machine learning models.
- **Amazon Lex** - Fully managed artificial intelligence (AI) service for building natural language models and conversational interfaces.

3.12 Miscellaneous Services

- **Amazon Polly** - Perform text-to-speech translation in multiple languages.

Chapter 4

Architecture Principles

4.1 Well-Architected Framework

The AWS Well-Architected Framework is designed to help you understand pros and cons to decisions about building systems in the cloud. The AWS Well-Architected Tool can be used to help review architecture against best practices. https://console.aws.amazon.com/wellarchitected

General Design Principles

- **Stop guessing capacity needs** - Cloud computing allows you to scale up or down capacity easily.
- **Test systems at production scale** - Spin up a production sized test environment as-needed to simulate the live environment. Tear down the environment when not in use to save money since AWS is pay for what you use.
- **Automate to make architectural experimentation easier** - Automation allows you to create and replicate workloads without manual effort. Track changes and audit the impact.
- **Allow for evolutionary architectures** - In a traditional environment architectures are usually implemented as static because change can be hard and expensive. The ability to automate and test in the cloud reduces the risk of design changes.
- **Drive architectures using data** - Collect data on how your architecture choices are impacting your workloads. You can then use decisions based on facts to improve how your workload performs.
- **Improve through game days** - Test your architecture by scheduling "game days" to simulate events in production.

Six Pillars

- **Operational Excellence** - This pillar focuses on running and monitoring systems, and continually improving processes/procedures.
  - *Perform operations as code* - Apply the same engineering discipline that you use for application code your entire environment.
  - *Make frequent, small, reversible changes* - Design application components to be updated regularly. Make changes in small increments that can be reversed if they fail.
  - *Refine operations procedures frequently* - Continually look for opportunities to improve operational procedures.

  - *Anticipate failure* - Perform "pre-mortem" exercises to identify potential sources of failure. Test scenarios to understand the impacts of those failures and see that your response procedures are adequate.
  - *Learn from all operational failures* - Apply lessons learned from all operational events and failures.
- **Security** - Focus on protecting information and systems.
  - *Implement a strong identity foundation* - Implement the principle of least privilege and enforce separation of duties. Centralize identity management.
  - *Enable traceability* - Monitor, alert, and audit actions in the environment in real time.
  - *Apply security at all layers* - Apply a defense in depth approach with multiple security controls at all levels.
  - *Automate security best practices* - Automate software-based security mechanisms and controls that are defined and managed as code.
  - *Protect data in transit and at rest* - Categorize data into sensitivity levels and apply encryption and other security controls appropriately.
  - *Keep people away from data* - Use tools and procedures to reduce or eliminate the need for manual processing of data.
  - *Prepare for security events* - Be prepared for an incident by having policies and plans for incident management and investigation.
- **Reliability** - Focus on workloads performing their intended functions, and the ability to recover quickly from failures.
  - *Automatically recover from failure* - Monitor workloads and key performance indicators to automatically trigger automation when a threshold is breached. This allows for automatic notification and either repair or workarounds for the issue.

  - *Test recovery procedures* - Perform testing to verify how the workload fails and validate recovery procedures.
  - *Scale horizontally to increase aggregate workload availability* - Replace a single large resource with multiple smaller resources to limit the impacts of a single failure.
  - *Stop guessing capacity* - Monitor workloads and automate the addition or removal of resources to maintain optimal level.
  - *Manage change in automation* - Changes to infrastructure should be changed using automation which can be tracked and reviewed.
- **Performance Efficiency** - Streamlined allocation of resources. Selecting resource types and sizes optimized for workload requirements.
  - *Democratize advanced technologies* - Make it easier for your team to implement new technologies easier by delegating complex tasks to the cloud vendor. Many technologies become managed services that your team can consume without resource provisioning or management.
  - *Go global in minutes* - Quickly deploy your workload in multiple regions around the world for lower latency and better customer experience.
  - *Use serverless architectures* - Serverless architectures remove th need to run and maintain your own servers.
  - *Experiment more often* - With virtual and automatable resources it is easier to test using different resource configurations.
  - *Consider mechanical sympathy* - Understand how cloud services are consumed and make sure the approach aligns with your workload goals.
- **Cost Optimization** - Focuses on avoiding unnecessary costs.
  - *Implement cloud financial management* - Dedicate time and resources to build capabilities for cost optimization.
  - *Adopt a consumption model* - Pay only for the computing resources you require. Development and test environments used only during certain hours / days can be stopped when not in use for cost savings.

  - *Measure overall efficiency* - Measure the business output of your software and the costs associated with delivering it.
  - *Stop spending money on undifferentiated heavy lifting* - AWS does the heavy lifting of data center operations and other infrastructure to allow you to focus on customers and business needs.
  - *Analyze and attribute expenditure* - The cloud makes it easier to accurately identity usage and costs of systems.
- **Sustainability** - Focuses on minimizing the environmental impact of running services.
  - *Understand your impact* - Measure the impact of your workload and model the future impacts.
  - *Establish sustainability goals* - Establish long-term sustainability goals such as reducing to compute and storage requirements per transactions.
  - *Maximize utilization* - Right-size workloads and implement efficient design to ensure high utilization / maximize energy efficiency of underlying hardware.
  - *Anticipate and adopt new more efficient hardware and software offerings* - Design for flexibility to allow for rapid adoption of new more efficient technologies.
  - *Use managed services* - Sharing services across a broad customer base helps maximize resource utilization and reduces the amount of physical infrastructure needed to support cloud workloads.
  - *Reduce the downstream impact of your cloud workloads* - Reduce the amount of resources required to use your services.

Chapter 5

Compute Services

5.1 Elastic Compute Cloud (EC2)

The Amazon Elastic Compute Cloud (EC2) provides compute resources (e.g. virtual servers) in the cloud. These EC2 instances are suitable for a wide range of use cases including: web application hosting, batch processing, API servers, desktop in the cloud, etc.

Each server created in EC2 (aka Instance) has a defined instance type consisting of processor, memory and storage. Instance type cannot be changed without downtime. Instances come in multiple categories: general purpose, compute optimized, memory optimized, storage optimized, or accelerated computing.

EC2 Storage Options

When creating an EC2 instance a root device type must be selected. This determines how the volume used to boot the instance will be stored. There are two root device types to choose from.

- **Instance Store** - Ephemeral storage that is maintained only while the instance is running. At boot the storage location is created and the image content is copied to the volume.
- **Elastic Block Store (EBS)** - Provides a persistent volume that is stored separately from the EC2 instance. This is the most commonly used storage mechanism for EC2.

Machine Images

EC2 instances are created based on Amazon Machine Images (AMI). An AMI provides a template for an EC2 instance including OS, configuration, and initial data. AWS provides many images to use as a starting point, or you can obtain images from a marketplace. Custom AWS images can be made and shared across accounts within an organization.

Purchase Options

- **On-demand** - Standard purchase method for EC2 instances. Billed per second while the instance is in the `running` state. No time based commitment required.
- **Reserved** - Reserved instances provide significant cost savings over on-demand instances, but they require a commitment to run the instance for a specific amount of time. This is not a dedicated server.
- **Savings Plan** - Similar to a reserved instance, but you are making a commitment measured in USD per hour.

- **Spot** - Significant cost savings for use cases that are flexible about when they are run and if they can be interrupted.
- **Dedicated** - Most expensive. Good if you have a per server licensing model to adhere to. Some types of compliance models need this.

5.2 Elastic Beanstalk

Amazon Elastic Beanstalk is a service that makes it easier to deploy applications on AWS. Supports specific technologies including Java, Node.js, Docker, and more. Leverages other AWS technologies. So, you only pay for the other services used and not for elastic beanstalk itself.

Elastic beanstalk allows for deploying applications with minimal knowledge of other AWS services. This is a lower maintenance service, but provides less customization options. Features include: monitoring, deployment, and scaling.

5.3 AWS Lambda

AWS Lambda is a serverless computing service that allows for application code to run without provisioning any server infrastructure. Commonly referred to as functions as a service. Only pay for what you use based on execution time and memory. Integrates well with event driven architectures and other AWS services. Benefits include: low maintenance, auto-scaling, fault-tolerance, servers architecture, and pay by usage.

5.4 Elastic Load Balancing

AWS elastic load balancing is a highly available and scalable service for managing traffic into an application. The service runs always in at least two availability zones and includes autoscaling based on load.

There are four types of load balancers:

- **Application Load Balancer (ALB)** - An application load balancer works at the application layer (OSI layer 7) and can load balance/proxy HTTP/HTTPS traffic. Allows for specifying advanced routing rules.
- **Network Load Balancer (NLB)** - A network load balancer works at the transport layer (OSI layer 4) and is capable of load balancing most TCP/UDP protocols.
- **Gateway Load Balancer** - Provides cloud native load balancing for virtual appliances.
- **Classic Load Balancer** - Older style of load balancer that supports TCP/SSL connections and EC2-classic. Supports sticky sessions using application-generated cookies.

Chapter 6

Storage & Database Services

6.1 Amazon Simple Storage Service (S3)

Amazon S3 is an object storage service that provides access over HTTPs. S3 stores objects (files) in buckets (like a directory).

Features

- High availability of objects across multiple availability zones.
- Options for replication either within the same region or across regions.
- Option to enable versioning of files per bucket.
- Static website hosting.
- Control access through IAM policies.

Buckets

- Must have a globally unique name (across regions and accounts) consisting of 3-36 lower case characters. Name must start with a letter or number and cannot contain underscores or be an IP address.
- Buckets are defined in a specific region even though it looks like a global service.
- S3 buckets have no concept of directories.

Objects

- Objects are identified by a key. Key is composed of a prefix and the object name.
- Objects can be up to 5TB.

Storage Classes

S3 storage is available with different storage options. These have varying cost, reliability, and access times. Option to enable intelligent-tiering to automatically move objects between tiers (for a fee).

- **Standard General Purpose** - Used for frequently accessed data.
- **Standard Infrequent Access (IA)** - Less frequent access, but still needed quickly.
- **One Zone Infrequent Access** - Only stored in one availability zone. Lower reliability, use for data that can be recreated easily.

- **Glacier** - Lower cost. Intended for archival or backup purposes.
  - *Glacier Instant Retrieval* - Must store for a minimum of 90 days. Millisecond retrieval.
  - *Glacier Flexible Retrieval* - Must store for a minimum of 90 days. Multiple retrieval options: Expedited 1 - 5 minutes, Standard 3 - 5 hours, Bulk 5 - 12 hours.
  - *Glacier Deep Archive* - Must store for a minimum of 180 days. Standard retrieval in 12hrs or bulk retrieval in 48hrs.

Data Migration

AWS provides multiple options for migrating data into AWS S3 to be used in other services.

Service	Transfer Capacity	Notes
Snowcone	8 - 14TB	Compact device. Can return to AWS for upload or upload yourself.
Snowball	80TB	Available in storage or compute optimized versions.
Snowmobile	100PB	Shipping container size, for massive data transfer.

6.2 Amazon Relational Database Service (RDS)

Amazon RDS is a fully-managed database service that makes it easy to operate a database in the cloud. RDS has support for multiple database server engines:

- MySQL / MariaDB
- PostgresSQL
- Oracle
- SQL Server
- Aurora

Key features of RDS include:

- Read replication for improved performance.
- Multi-AZ and Multi-Region deployment options for availability.
- Runs on EC2 instances, but you do not have access to the instances. AWS is fully responsible for the security and management of the database server instance.
- Support for automatic backups. Automatic backups are performed daily and transaction logs are retained throughout the day. This allows for restoration to a very specific time if needed. **These backups are deleted when the RDS instance is deleted.**
- Support for DB snapshots performed manually by the administrator. These are retained even if the RDS instance is deleted.