

# AWS Migration Services

**NOTE:** This content will be available as a video course soon! We are making it available as a Learning Path resource in the meantime to help you prepare for the newly updated AWS Certified Solutions Architect - Associate (SAA-C03) exam.

## Introduction

Hello and welcome to this short course that will give you an introduction to some of the AWS migration services that are available to help you migrate your solutions and data from your on-premise environment onto the AWS cloud.

My name is Stuart Scott, and I am the AWS content Director here at Cloud Academy. Feel free to connect with me to ask any questions using the details shown on the screen, alternatively you can always get in touch with us here at Cloud Academy by sending an e-mail to [support@cloudacademy.com](mailto:support@cloudacademy.com) where one of our Cloud experts will reply to your question.

## Who should attend this course?

This course has been created for those who are:

- Looking to migrate services and applications from their own on-premise environment to the AWS Cloud
- Also, this course will be advantageous to anyone looking to sit the one of the Solution Architect certifications

## Learning Objectives

By the end of this course you will have a greater understanding of:

- The 3 stages of the migration process
- The AWS services associated with each stage
- A foundational understanding of each service related to an AWS migration

## Prerequisites

You should have a basic understanding of AWS architecture and some of the common AWS services when building solutions.

## Feedback

Feedback on our courses here at Cloud Academy is valuable to both us as trainers and any students looking to take the same course in the future. If you have any feedback, positive or negative, it would be greatly appreciated if you could contact [support@cloudacademy.com](mailto:support@cloudacademy.com).

Please note that, at the time of writing this content, all course information was accurate. AWS implements hundreds of updates every month as part of its ongoing drive to innovate and enhance its services.

As a result, minor discrepancies may appear in the course content over time. Here at Cloud Academy, we strive to keep our content up to date in order to provide the best training available. So, if you notice any information that is outdated, please contact [support@cloudacademy.com](mailto:support@cloudacademy.com). This will allow us to update the course during its next release cycle.

Thank you!

## The 3-stage approach to migration

When planning a migration from your own on-premise data center to AWS, it can be more effectively managed and implemented if looked at from a 3-stage perspective, and AWS has categorized its migration services to help align to this process. You may or may not use each service, however it provides a great starting point to help you plan your migration.

The stages that have been defined by AWS consist of:

1. Assess
2. Mobilize
3. Migrate and Modernize

The first stage, **Assess**, focuses on the start of your journey and forms the basis of understanding your current environment to determine how prepared and ready your IT infrastructure is to move to the AWS. This will help you to formulate the goals and objectives of your migration and allow you to present an effective business case to your leadership team.

The migration services related to this stage of the process are:

- Migration Evaluator
- AWS Migration Hub

Stage 2, **Mobilize** has more of an emphasis on defining details of your migration plan and your strategy, it will also help you to iron out any kinks in your business plan with regards to the

potential requirements of needing to perform specific migration strategies, which usually form 1 of the these 7 methods for your applications:

- Relocate
- Rehost
- Replatform
- Refactor
- Repurchase
- Retire
- Retain

With this in mind, you have the ability to understand your baseline environment in addition to identifying any skill gaps you may have within your organization.

The migration services related to this stage of the process are:

- AWS Application Discovery Service
- AWS Control Tower

Next we have stage 3, **Migrate and Modernize**. This is when you can design your application solution to run on AWS, understanding the services required, the interconnections between different services required when migrated, in addition to the validation of the design. Depending on which of the 7 migration strategies used for each application, will ultimately depict which AWS service you should use to migrate your application and its associated data.

The migration services related to this stage of the process are mixed between those required to migrate servers, databases and applications which are:

- AWS Application Migration Service
- AWS Database Migration Service

...and then those services which are more oriented towards managing the migration of data, which include:

- AWS DataSync
- AWS Transfer Family
- AWS Snow Family
- AWS Service Catalog
- AWS Storage Gateway

## ‘Assess’ - Understanding your environment

In this lecture I want to provide you with an understanding of what the **Migration Evaluator** and **AWS Migration Hub** services are in greater detail to help you ascertain when and why you might use either of these services during the assessment stage of your migration plan.

### Migration Evaluator

Migration Evaluator was formerly known as TSO Logic, however in 2018 AWS acquired TSO Logic and the Migration Evaluator service was born.

Being associated with the ‘assess’ step of your cloud migration, you can expect that Migration Evaluator has an element of data gathering. It provides a mechanism to help you to understand and implement a baseline of what your environment looks like on premise and how this could be projected to run in AWS with associated cost modeling through data analysis. This analysis is helpful to speed up your journey to a successful digital transformation onto the AWS Cloud platform.

The Migration Evaluator focuses primarily on understanding your compute, storage and Microsoft licenses that you are currently using and automatically helps to identify which resource options would provide the best cost optimized solution in AWS for each given workload that you are operating. This right sizing exercise for EC2 helps you to keep your expenditure low when undergoing a migration and these recommendations are determined through an algorithm that analyzes your hardware's CPU utilization, the age of the processor, the available RAM, in addition to the OS being run and its Microsoft license type. When compared to the costs of running your environment on premises, Migration Evaluator can help to reduce costs up to 50% when running in AWS.

To be able to perform this analysis, Migration Evaluator has an agentless collector tool that gathers the relevant data to suggest its recommendations, including inventory discovery and time-series performance data. Alternatively, these metrics and data can also be ingested by the Migration Evaluator using other 3rd party tools. It is recommended that this data collection and gathering exercise is run for at least 2 weeks. Once collected, this data can then also be read by other AWS services such as the AWS Migration Hub once it has been collected/received by the Migration Evaluator.

Once the data has been received, the Migration Evaluator will be able to present Quick Insights document in a few hours, which showcase the results of the assessment and evaluation highlighting recommendations and also the projected and expected costs after the migration, in addition to identifying any issues relating to the scope of resources required based upon your workloads.

Following the result presented in the Quick Insights documents, you can request further analytical information in the form of a directional business case, and an Analytics engine, but this can take up to a further 7 days to receive it from the AWS Migration Evaluator team. These help to dive into and analyze different cost modeling scenarios and a full inventory of compute and storage resources.

If you would like to request an assessment by the Migration Evaluator then you can either speak with your AWS account manager to organize it, alternatively you can visit

<http://aws.amazon.com/migration-evaluator/>

## AWS Migration Hub

The AWS Migration Hub provides a dashboard view of your migration project, and so, it's a very useful way to plan, track and manage a migration project. It is essentially the nerve center of your migration enabling you to discover and migrate services you have been running in identified locations.

A migration project can often include a wide variety of services and servers often located in a number of different data centers or facilities. You might have databases located in one data center and business applications located in various points in another office or co-location. There may also be different versions and configurations of service, which will need to be taken into account if you go to migrate through a different platform. Now one of the challenges with migrations is maintaining one view of the various aspects of this project. So the AWS Migration Hub is a very powerful tool for large migrations where there are multiple locations or multiple servers and services. So how does it work?

Well, as the name implies, the Amazon Migration Hub is the central hub of your migration project. It is accessed and run from the AWS Management Console, and once you have activated the Migration Hub, you can use it to discover and manage your migrations. It integrates with other AWS services to discover and audit your server inventory on a number of different networks and nodes, and it can also manage and track the migration of services and servers and applications to AWS Cloud Services.

The Migration Hub receives and collects server inventory from a variety of different sources, these include:

- **Migration Hub import** – This allows you to import data from your own on-premises servers and applications.
- **Migration Evaluator Collector** – As already discussed, the Migration Evaluator can be used to feed into Migration Hub providing insights to help build a business case and plan strategy.

- **AWS Agentless Discovery Connector** – This a VMware appliance that is used to collect data and discover information relating to any VMware virtual machines that you might be using within your data center.
- **AWS Application Discovery Agent** – This Agent, created by AWS, is installed on your own servers and virtual machines that you would be running in your current data center with the main objective being to collate metrics and information relating to system configuration, system performance, running processes, and network connections.

Using these sources, the Migration Hub can then view technical specifications and performance information about the discovered resources. With discovery, we can get an in depth analysis of what it would look like, how many servers we have, how many applications we have, we can group them together, and perform detailed analysis.

When it comes to the actual migration, you can choose a migration service tool that can be selected from the AWS Migration Hub. If you are looking to migrate a server, then you can use the **AWS Application Migration Service**, if you want to migrate a database, then you would need to select the **AWS Database Migration Service**.

## ‘Mobilize’ - Migration and strategy planning

The Mobilize stage of planning a migration to AWS has a focus on two key AWS services, these refer to the:

- AWS Application Discovery Service
- AWS Control Tower

In this lecture, I want to explain what these services are used for enabling you to determine when and if you need them as a part of your migration planning.

### AWS Application Discovery Service

So, the AWS Application Discovery Service follows on nicely from the Assess stage, and enables you to dive deeper into your workloads by collating additional information, such as usage, configuration, and behavior data from these workloads running within your data centers. With this useful information collated, security is of course paramount and so the Application Discovery Service encrypts all information that it has retained. This encrypted data can also be read by other AWS services, such as the AWS Migration Hub. Also, if you are familiar with Amazon Athena and Amazon QuickSight you can export the data for additional analysis and help you identify the TCO of running your workloads in AWS.

So how does the Application Discovery Service gather key data from your resources in your own data centers? Well it can be done in one of two different ways.

Firstly, through an Agent-based discovery, and secondly via an Agentless discovery.

### Agent-based discovery

Let's look at the agent-based option first. As expected, in this scenario an agent is installed across your fleet of servers that you want to gather information for, which can be installed on both Linux and Windows operating systems. Also, this agent can be installed both on physical servers in addition to virtual machines. When installed, the agent is then used to capture configuration data, system performance, network connections, in addition to the different processes that are running on the server, helping you to map out service dependencies.

Once the agent has been registered, it will connect to the AWS Application Discovery Service in the specified region and integrate with the AWS Migration Hub. From this point, the agent will gather the data and send it back to the application Discovery Service over a secure connection using Transport Layer Security (TLS) approximately every 15 minutes. For an in-depth look at the data captured, please refer to the AWS documentation found here:

<https://docs.aws.amazon.com/application-discovery/latest/userguide/agent-data-collected.html>

### Agentless Discovery

Now let's look at the Agentless discovery, which is also referred to as the Discovery Connector. The first thing to mention is that this option is only available for gathering data on your VMware virtual machines. The reason being is that agentless discovery is configured by deploying an AWS agentless Discovery Connector (OVA file) via your VMware vCenter. Again, it captures information relating to each VM and which vCenter they are associated with, in addition to configurational data, IP and MAC address data, utilization and disk resource allocations, plus average peak metrics for RAM, CPU and Disk I/O.

Much like the Agent-based discovery, it operates in a very similar way. Once the connector has been distributed as the OVA file, it will connect to the AWS Application Discovery Service and integrate with the AWS Migration Hub. However, communication between the Connector and the Application Discovery Service will only occur approximately once every 60 minutes. Also, as expected any data collected is sent over an encrypted TLS connection back to the service.

### AWS Control Tower

AWS Control Tower is an essential service if you are looking to deploy a multi-account strategy as a part of your migration. It provides a simple and effective way to set up your accounts, teams, in addition to meeting governance requirements based upon best practices in the form of a Landing Zone, but what is a landing zone?

A Landing Zone is a multi-account architecture that follows the well-architected framework and is based around the ideas of security and compliance best practices. Your landing zone will be

automatically created by AWS Control Tower and it is inherent part of the service, your landing zone is created from a series of best practice blueprints that help us set up systems that deal with identity, federated access and overall account structure, these blueprints do the following on your behalf, they create a multi-cloud environment using AWS organizations.

There are three organizational units that are provisioned here, the Root OU, this will be the parent OU that contains every other OU, within your landing zone, the core OU this OU contains the log archive and any audit member accounts, we generally refer to these accounts as the shared accounts and a custom OU, this OU another member OUs contain the actual working accounts that your users need to perform whatever duties they do within your AWS environment, for example your developer AWS accounts would sit within this OU.

The service also builds out two shared accounts, a log archive in the account which will be the place where all the logs will be sent between all accounts, it will store the logs of all API calls and resource configurations for every account within the landing zone and an audit account which has a restricted account that has been created to give your security and compliance team members read and write access to any account within your AWS landing zone.

From this account you'll have programmatic access to review all other accounts, by way of a role that grants use of Lambda functions only, this account does not allow you to log into other accounts manually. Control tower will also provide identity management with the use of AWS single sign-on, default directory, this directory will house all of the AWS SSO users, you can also use it to define the scope or permissions available for each of those users. It will also provide federated access to those accounts, using AWS SSO directly, control tower then hooks up centralized logging from AWS cloud trail, and into its config, which is stored securely within AWS three and the logging account.

Finally it enables cross account security auditing using AWS IAM and AWS SSO to allow the audit account to perform routine checks as it wishes. In the end AWS Control Tower will have all of that automatically set up for you. That is incredible, the amount of time it might take you to create all that by hand would probably be measured in weeks if done by a single person. The service can even create pre-configured groups such as your admins, users, and auditors, you can of course create more designations that provide different levels of access based on your needs.

Additionally, if you have your own active directory service already going, you can plug that into AWS SSO and configure it to work directly with your system which again can be a cloud-based AD or when you're hosting on-premises, just make sure to use AD connector or the AD service.

When you create your new landing zones using AWS Control Tower, there are a large number of AWS resources that are created on your behalf, you need to be very careful about deleting or removing these pre-configured resources. If you were to destroy or tamper with these resources you can send the control tower into an unknown state.



# 'Migrate and Modernize' - Designing your solutions on AWS

In this lecture I want to cover some of the services that can be used to help you manage the migrate and modernize stage of your migration strategy, so the point at which you are actually moving your solutions and data into AWS. This lecture will be split into two different sections, firstly, I shall look at the services that help you to migrate your servers, database and applications, and then I shall focus on how to migrate data from your data center to the AWS cloud.

## AWS Application Migration Service

The AWS Application Migration Service is a great service to help you migrate your applications to AWS with minimal downtime and interruption, such as those running SAP, Oracle, and SQL Server. The Application Migration Service is the suggested and recommended solution when migrating your applications over using CloudEndure.

The service simply works on the basis of a lift-and-shift approach by converting your existing physical and virtual machines to running natively across AWS infrastructure using an agent which replicates your source servers to virtual instances in AWS while the source server continues to run. This simplified approach helps to keep cost to a minimum and helps to simplify the migration process.

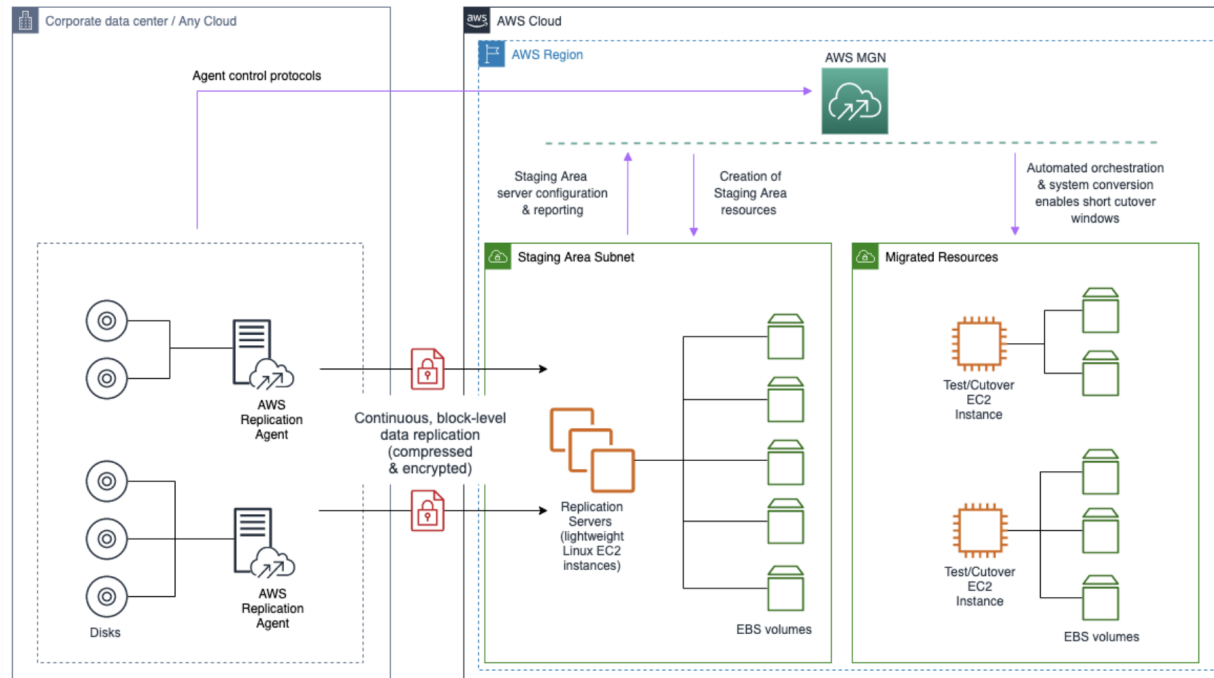
Once your servers have been migrated, you can leverage the AWS cloud to further optimize your infrastructure through either refactoring or replatforming your applications.

To begin the migration, you will first need to configure a *replication settings template* which will help you to configure how data replication will be managed from each source server. You will need to specify settings for a replication server which is used to replicate data between your source servers on premise and instances in AWS, which will include:

- The staging area subnet
- Instance type
- EBS Volume type
- EBS encryption
- Security Groups
- Data routing and throttling
- Resource tags

Once this template is created, you can then add your source servers, either Windows or Linux,, which is completed by installing an AWS replication agent. Once your agent is installed on your physical or virtual machines, it will appear within the Application Migration Service and will follow a Migration workflow and will display its status of each stage of the migration.

This image here shows how the agent integrates with the staging environment using your replication settings template, before being tested and cutover to being fully migrated resources.



Source: <https://docs.aws.amazon.com/mgn/latest/ug/adding-servers-gs.html>

Launch settings within the service can be used to help you configure how each source server will be Tested before Cutover to a migrated server. Here you will be able to configure settings such as:

- Instance type right-sizing
- Private IP addressing
- The transfer of server tags
- OS Licensing
- And if you want the instance to launch once it's cutover or if you want to start it manually

Of course testing is an integral part of this service, you need to be sure that your application will be running as expected before being cutover into your production environment. Testing can be managed on an individual server basis, or as a group of servers. Once your servers have been successfully tested, they can then be launched as a Cutover instance in your production environment based on the launch settings previously specified.

## AWS Database Migration Service

As you probably expect from the name, this service has been designed to help you migrate your relational, NoSQL databases, and data warehouses, which could be from your own environment to

AWS, with minimal downtime and security in mind. This is a very effective way to migrate your on premise databases from an array of commercial and open-source databases.

The Database migration service is very flexible in terms of its capabilities from a migration standpoint. You can migrate from and to the same database, such as Oracle to Oracle, but you can also migrate from different source to destination databases, making use of some of AWS's most cost effective DB solutions. An example of this would be to migrate from an on premise database such as SQL Server, and migrating your data to the AWS service of Amazon Aurora, which is designed for very high performance and availability.

You may also have scenarios where by you are looking to consolidate your database environments, using the Database Migration service you can move multiple database workloads into a single Amazon Redshift environment allowing you to scale to a petabyte-sized data warehouse, providing you with an opportunity to gain insights and analyze your data.

If you are looking to migrate your database to a target database engine that is compatible with one another then the migration process is simplified. If components of the database share the same schema structure, data types, and code then the operation becomes more of a single step process making the DMS service very efficient.

However, if the source and target database engines that you have selected are different, and you might want to do this to make use of the capabilities of the different AWS database services that are available, then additional steps will be required. For example, let's say your schemas between the source and target operate differently, then you will need to perform a schema transformation before the migration can take place using the AWS Schema Conversion Tool (AWS SCT).

When running a migration, you must specify your source and destination database targets. The DMS service will then create a Replication instance, which is effectively an EC2 instance with one or more replication tasks. Endpoints are also created to connect to your source datastore and your destination target datastore. Replication tasks are then run from your replication instance to move data sets from your source to your destination targets.

Ok, so now we have covered the first section, I can now move on to the next section where we look at the migration of data, starting with AWS DataSync.

## AWS DataSync

If you've been working with different AWS storage services for any length of time then you may have already come across this service. AWS DataSync is a service that allows you to easily and securely transfer data from your on-premise data center to [AWS](#) storage services. It can also be used to manage data transfer between 2 different AWS storage services too, so it's a great service to help you migrate, manage, replace and move data between different storage locations.

At the time of writing this course, [AWS DataSync](#) supports the ability to work with data stored on Network File Systems shares, Server Message Block shares, and any self-managed object storage, in addition to the following AWS services:

- Amazon S3
- Amazon Elastic File System
- Amazon FSx for Windows File Server
- AWS Snowcone

When performing data transfer operations, whether this be from on-premises or between AWS storage services, DataSync support AWS VPC Endpoints and so its able to utilize the high bandwidth, low latency AWS network to it's advantage, this helps to both simplify the management of the request and automate your data transfer across secure infrastructure. For more information on AWS VPC Endpoints, please see our AWS Networking lecture found here:

### VPC Endpoints

<https://cloudacademy.com/course/aws-networking-features-essential-for-a-solutions-architect/vpc-endpoints/>

With data transfer speed a being a key factor for a data transfer services, AWS Data Sync comes with its own purpose-built data transfer network protocol in addition to a parallel and multithreaded architecture to rapidly perform data transfer, this means that each DataSync task has the potential of utilizing 10 Gbps over a network link between your own on-prem data center and your AWS environment.

Obviously when working with data, especially when moving it between 2 points, security is a key concern. As a result AWS DataSync provides 2 levels that provide end-to-end security, these being encryption, in addition to data validation.

From an encryption perspective, encryption in transit is implemented by encrypting the data using the Transport Layer Security (TLS) protocol. When data reaches an AWS service, it also supports encryption at rest mechanisms that EFS and FSx for Windows service offers, in addition to the default encryption at rest option for Amazon S3.

The 2nd point, Data Validation, ensures that your data arrives at its destination in one piece, exactly as it was when it left the source ensuring that it wasn't compromised or damaged in any way during its transit. This additional check helps to validate the consistency of your data that was written to the AWS storage service, and that its a perfect match from when it left its source location.

From a cost perspective, AWS DataSync uses a flat pricing strategy based on a per-gigabyte of data transferred, this makes it easy to predict avoiding any unexpected costs.

## AWS Transfer Family

The AWS Transfer Family is specifically designed to help you securely transfer data both into and out of two of the most commonly used storage services that AWS has to offer, these being Amazon S3 and the Elastic File System (EFS). To learn more about Amazon S3 and Amazon EFS, please refer to the following content:

### Amazon S3: Deep Dive

<https://cloudacademy.com/learning-paths/deep-dive-amazon-s3-955/>

### Using EFS to Create Elastic File Systems for Linux-based Workloads

<https://cloudacademy.com/course/using-amazon-efs-to-create-elastic-file-systems-for-linux-based-workloads/>

This file transfer can be completed using one of four supported protocols:

1. Secure Shell File Transfer Protocol, more commonly known as SFTP, providing encryption over SSH
2. File Transfer Protocol Secure, referred to as FTPS and uses TLS encryption
3. File Transfer Protocol, FTP, which is an unencrypted connection
4. Applicability Statement 2, known as AS2

Being a fully managed service, this transfer of data is enabled without you having to provision any of your own server infrastructure, instead the AWS Transfer Family utilizes its own file transfer protocol-enabled instance instead simplifying the process and reduces the need to alter the configuration of your applications. It is also a highly available service, operating in up to 3 different availability zones with the additional support of being backed by auto scaling to ensure your transfer requests are met without an issue.

The Transfer Family utilizes a Managed File Transfer Workflow (MFTW) which enables you to configure, run and implement a level of automation to help you manage your file transfers allowing you to track the process of the transfer from beginning to end. By utilizing MFTW you can configure specific processing actions that you might want to take prior to transferring your data, such as tagging, copying, enabling encryption, and filtering, plus other common file-processing actions.

To begin using the AWS Transfer family there are a number of steps to take.

- Of course, you must first have your destination storage configured and available, which will either be an S3 bucket, or an EFS File System.
- Next, and through the use of IAM roles, you must grant the required permissions to the storage destination for the AWS Transfer Family.

- Once your storage destination is set up with the appropriate permissions applied, you can then configure a Transfer Family Server using one of the support protocols to manage your file transfers to that storage destination. As a part of this configuration you can also implement optional CloudWatch logging metrics to help you monitor your transfer requests.
- You must then add a User to the Transfer Server and associate it with the role you created previously allowing access to your S3 bucket of Elastic File System
- You are then ready to complete the transfer using a Client, such as OpenSSH, WinSCP, Cyberduck or FileZilla. For additional information on how to do this for each client, please visit the following URL:  
<https://docs.aws.amazon.com/transfer/latest/userguide/transfer-file.html>

## AWS Snow Family

When talking about the AWS Snow Family, I want to answer 2 simple questions:

1. What is the snow family
2. and what does it consist of?

So firstly, what is it? The snow family consists of a range of physical hardware devices that are all designed to enable you to transfer data into [AWS](#) from the edge or beyond the Cloud, such as your Data Center, but they can also be used to transfer data out of AWS too, for example, from Amazon S3 back to your Data Center.

It's unusual when working with the cloud to be talking about physical devices or components, normally your interactions and operations with AWS generally happen programmatically via a browser or command line interface. The snow family is different, instead, you will be sent a piece of hardware packed with storage and compute capabilities to perform the required data transfer outside of AWS, and when complete, the [device](#) is then sent back to AWS for processing and the data uploaded to Amazon S3.

You can perform data transfers from as little as a few terabytes using an AWS snowcone all the way up to a staggering 100 petabytes using a single AWS snowmobile, and I'll be talking more about these different snow family members shortly. Now of course when we are talking about migrating and transferring data at this magnitude, using traditional network connectivity is sometimes simply not feasible from a time perspective. For example, let's assume you needed to transfer just 1petabyte of data over a 1gbps using Direct Connect it would take 104 Days, 5 Hours 59 Minutes, 59.25 Seconds, not forgetting the cost of the data transfer fees too!

In addition to these devices packing some serious storage capacity for data transfer, some of them also come fitted with compute power, allowing you to run usable EC2 instances that have been designed for the snow family enabling your applications to run operations in often remote and difficult to reach environments, even without having a data center in sight, and when working with a lack of persistent networking connectivity or power. For example, the snowcone comes with the

ability to add battery packs increasing their versatility. The enablement of running EC2 instances makes it possible to use these devices at the edge to process and analyze data much closer to the source.

So let's now take a look at what the Snow family consists of to get a better understanding of what these devices are.

	 AWS SNOWCONE	 AWS SNOWBALL COMPUTE OPTIMIZED	 AWS SNOWBALL COMPUTE OPTIMIZED WITH GPU	 AWS SNOWBALL STORAGE OPTIMIZED	 SNOWMOBILE
HEIGHT	3.25"	19.75"	19.75"	19.75"	9.6'
WIDTH	5.85"	12.66"	12.66"	12.66"	8'
LENGTH	8.94"	21.52"	21.52"	21.52"	45'
WEIGHT	2.1kg	21.3 kg	21.3kg	21.3kg	68,000 lbs
vCPU	2	52	52	24	N/A
MEMORY	4GB	208GB	208GB	32GB	N/A
STORAGE HDD	8TB	39.5TB	39.5TB	80TB	100PB
STORAGE SSD	N/A	7.68	7.68	N/A	N/A
GPU	N/A	N/A	NVIDIA Tesla V100 GPU	N/A	N/A

As you can see from this table, both from a physical and capacity perspective, the snowcone is the smallest followed by the snowball and finally the snowmobile. You may also notice that the snowball comes in 3 choices, compute optimized, compute optimized with GPU, and storage optimized, each offering a different use case, however, each of these 3 offerings all come in the same size device.

## AWS Service Catalog

AWS service catalog is an organizational tool developed with the purpose of making provisioning and creation of IT stacks easier for both the end user as well as your IT admins.

These stacks can include almost everything under the AWS sun – such as EC2 instances, Databases, software, and all the underlying infrastructures to create multi-tiered applications and architectures.

Service Catalog allows your end users to select the content that they need from a list of preapproved services that your IT or Admin teams set up ahead of time. This helps to bring down those barriers of entry for content creation, as well as helping to keep best practices and system security a key component of any deployment and migration.

With the ability to browse through a list of 'Products' which are just a set of pre-approved services, we can create and build with the full confidence that we as developers are creating solutions using only acceptable components that our security, administration, and leadership teams approve of.

Products are an IT service that you want to make available for deployment on AWS. A product can consist of just a single AWS Resource or can consist of multiple items such as EC2 instances, their associated EBS volumes, Database that you want them connected to, and all the monitoring capabilities you would come to expect from these services within the cloud.

A product can even be a package listed on the AWS Marketplace. For example, this could be helpful if you were using a database that AWS does not natively support but was available on the marketplace. In the end, a product is a service. It can range from something as small as a single instance doing basic web hosting to an enormous multi-tiered web application.

AWS service catalog requires you to upload AWS Cloud Formation Templates, and from these templates, the service will add that entire stack into the catalog as a single product. Again that product can be something as small as a single EC2 instance, or a very large multi-tiered web application.

Once your products are created, you can add these to AWS Service Catalog Portfolios. A portfolio is a collection of products with configuration information that helps in determining who can use the products within. Each portfolio requires a name, description, and a product owner. That last one is very important because if something goes wrong with an available product it's important to know who to send your complaints to.

You can also share portfolios between other AWS accounts, and give the administrators of those accounts the ability to add their own products to your portfolio. This could be useful when you have teams that operate independently, that each deal with creating their own products.

One of the best features and really the whole goal of service catalog, is that you have full control over what your end users have access to. From an administrator perspective that is incredibly powerful and can help you maintain high levels of both security and credibility.

AWS Service Catalog allows you to apply constraints on the products within your portfolios. These constraints allow you to limit the scope and ability of your products based on predefined settings. They also allow you to have additional functionality, at least on the administrative side.



## AWS Storage Gateway

Storage Gateway allows you to provide a gateway between your own data center's storage systems such as your SAN, NAS or DAS and Amazon S3, Glacier, and Amazon FSx on AWS.

The Storage Gateway itself is a software appliance that is stored within your own data center which allows integration between your on-premise storage and that of AWS. This connectivity can allow you to massively scale your storage requirements both securely and cost efficiently. The software appliance can be downloaded from AWS as a virtual machine which can then be installed on your VMware or Microsoft hypervisors.

Storage Gateway offers different configurations and options allowing you to use the service to fit your needs. It offers file, volume and tape gateway configurations which you can use to help with your DR and data backup solutions.

### File gateways

File gateways allow you to securely store your files as objects using File Gateway for Amazon S3 or you can use File Gateway configuration on Amazon FSx File Gateway, which provides access to in-cloud Amazon FSx for Windows File Server shares.

Using the S3 File Gateway allows you to map drives to an S3 bucket as if the share was held locally on your own corporate network. When storing files using the file gateway they are sent to S3 over HTTPS and are also encrypted with S3's own server side encryption SSE-S3.

In addition to this, a local on-premise cache is also provisioned for accessing your most recently accessed files to optimize latency which also helps to reduce egress traffic costs. When your file gateway's first configured you must associate it with your S3 bucket which the gateway will then present as a NFS V.3 or V4.1 file system to your internal applications.

This allows you to view the bucket as a normal NFS file system, making it easy to mount as a drive on Linux or map a drive to it in Microsoft. Any files that are then written to these NFS file systems are stored in S3 as individual objects as a one to one mapping of files to objects.

### Volume Gateways

Volume Gateways can be configured in one of two different ways, Stored volume gateways and cached volume gateways. Let me explain stored volume gateways first.

**Stored volume gateways** are often used as a way to backup your local storage volumes to Amazon S3 whilst ensuring your entire data library also remains locally on-premise for very low latency data access. Volumes created and configured within the storage gateway are backed by Amazon S3 and are mounted as iSCSI devices that your applications can then communicate with.

During the volume creation, these are mapped to your on-premise storage devices which can either hold existing data or be a new disk. As data is written to these iSCSI devices the data is actually written to your local storage services such as your own NAS, SAN or DAS storage solution. However the storage gateway then asynchronously copies this data to Amazon S3 as EBS snapshots.

Having your entire dataset remain locally ensures you have the lowest latency possible to access your data which may be required for specific applications or security compliance and governance controls whilst at the same time providing a backup solution which is governed by the same controls and security that S3 offers. Storage volume gateways also provide a buffer which uses your existing storage disks. This buffer is used as a staging point for data that is waiting to be written to S3.

**Cached volume gateways** are different from stored volume gateways in that the primary data storage is actually Amazon S3 rather than your own local storage solution. However cache volume gateways do utilize your local data storage as a buffer and the cache for recently accessed data to help maintain low latency, hence the name, Cache Volumes.

Again, during the creation of these volumes they are presented as iSCSI volumes which can be mounted by an application service. The volumes themselves are backed by the Amazon S3 infrastructure as opposed to your local disks as seen in the stored volume gateway deployment. As a part of this volume creation you must also select some local disks on-premise to act as your local cache and a buffer for data waiting to be uploaded to S3.

Although all of your primary data used by applications is stored in S3 across volumes, it is still possible to take incremental backups of these volumes as EBS snapshots.

The final option with AWS Storage Gateway is a tape gateway known as Gateway VTL. Virtual Tape Library. This allows you again to back up your data to S3 from your own corporate data center but also leverage Amazon Glacier for data archiving. Virtual Tape Library is essentially a cloud based tape backup solution replacing physical components with virtual ones.

This functionality allows you to use your existing tape backup application infrastructure within AWS providing a more robust and secure backup and archiving solution.

## Summary

In this course I wanted to cover some of the different services that can be used during an AWS Migration project to help you manage and plan your journey successfully. I began by introducing the 3 stage approach to migrations, these being:

- Assess
- Mobilize
- Migrate and Modernize

In the following lectures I then dived into each of these stages a bit further to identify which AWS services typically fall under each stage of your migration.

We learnt the key differences between each of the following services, and as a quick recap, let's review each of them in a sentence or two:

The **AWS Migration Evaluator** helps you to identify your compute, storage and Microsoft licenses that you are currently using and automatically helps to identify which resource options would provide the best cost optimized solution in AWS for each given workload that you are operating.

The **AWS Migration Hub** provides a dashboard view of your migration project, and so, it's a very useful way to plan, track and manage a migration project. It is essentially the nerve center of your migration enabling you to discover and migrate services you have been running in identified locations.

The **AWS Application Discovery Service** enables you to dive deeper into your workloads by collating additional information, such as usage, configuration, and behavior data from these workloads running within your data centers. This data can also be read by other AWS services, such as the AWS Migration Hub, Amazon Athena and Amazon Quicksight for additional analysis

**AWS Control Tower** is an essential service if you are looking to deploy a multi-account strategy as a part of your migration. It provides a simple and effective way to set up your accounts, teams, in addition to meeting governance requirements based upon best practices in the form of a Landing Zone

The **AWS Application Migration Service** is a great service to help you migrate your applications to AWS with minimal downtime and interruption, such as those running SAP, Oracle, and SQL Server. Using a lift-and-shift approach it converts your existing physical and virtual machines to running natively across AWS infrastructure using an agent which replicates your source servers to virtual instances in AWS while the source server continues to run.

The **AWS Database Migration Service** has been designed to help you migrate your relational, NoSQL databases, and data warehouses, which could be from your own environment to AWS, with minimal downtime and security in mind. It's very flexible in terms of its capabilities from a migration standpoint. You can migrate from and to the same database in addition to migrating from different source and destination databases.

**AWS DataSync** is a service that allows you to easily and securely transfer data from your on-premise data center to AWS storage services. It can also be used to manage data transfer between 2 different AWS storage services too, so it's a great service to help you migrate, manage, replace and move data between different storage locations.

The **AWS Transfer Family** is specifically designed to help you securely transfer data both into and out of two of the most commonly used storage services that AWS has to offer, these being Amazon S3 and the Elastic File System (EFS).

The **AWS Snow Family** consists of a range of physical hardware devices that are all designed to enable you to transfer data into AWS from the edge or beyond the Cloud, such as your Data Center, but they can also be used to transfer data out of AWS too, for example, from Amazon S3 back to your Data Center.

The **AWS Service Catalog** is an organizational tool developed with the purpose of making provisioning and creation of IT stacks easier for both the end user as well as your IT admins. These stacks can include almost everything under the AWS sun – such as EC2 instances, Databases, software, and all the underlying infrastructures to create multi-tiered applications and architectures.

The **AWS Storage Gateway** allows you to provide a gateway between your own data center's storage systems and Amazon S3, Glacier, and Amazon FSx on AWS. The Storage Gateway itself is a software appliance that is stored within your own data center which allows integration between your on-premise storage and that of AWS. This connectivity can allow you to massively scale your storage requirements both securely and cost efficiently.

That now brings me to the end of this lecture and to the end of this course, and so you should now have a greater understanding of some of the different AWS services that are available to help your through your AWS migration journey.

Feedback on our courses here at Cloud Academy is valuable to both us as trainers and any students looking to take the same course in the future. If you have any feedback, positive or negative, it would be greatly appreciated if you could contact [support@cloudacademy.com](mailto:support@cloudacademy.com).

Thank you for your time and good luck with your continued learning of cloud computing. Thank you.



**Stuart Scott, AWS Content Lead**

[stuart.scott@cloudacademy.com](mailto:stuart.scott@cloudacademy.com) | [@Stuart\\_A\\_Scott](https://twitter.com/Stuart_A_Scott) | [uk.linkedin.com/in/stuartanthonymcscott](https://uk.linkedin.com/in/stuartanthonymcscott)