Storage Migration to AWS

Technical White Paper

October 2018

Contents

Abstract

One of the most challenging steps in moving workloads to the cloud is to determine the best method for moving the different types of data storage utilized by enterprise applications. This paper will illustrate examples of migration to Amazon Web Services (AWS) for three types of storage: Network Attached Storage (NAS) file shares, host-attached block storage, and virtual disks. This paper will highlight the unique challenges of migrating these different storage types as well as present tools and solutions offered by AWS partners to efficiently transition the data to cloud.

Introduction

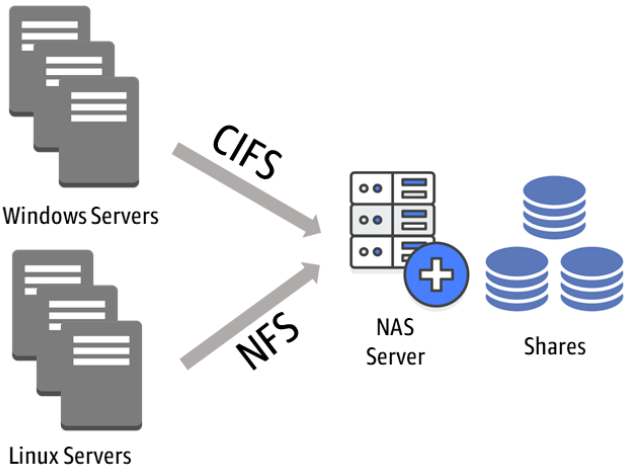
This paper will focus on three of the most common storage use cases in the enterprise today – NAS file shares, host-attached block storage, and virtual disks. In this paper, we will cover the Amazon Web Services (AWS) Partner Network (APN) partner solutions that provide automation and orchestration to not only replicate data to AWS, but enable application and workload consistency after the data migration is complete.

In planning the migration, block storage data might persist to Amazon Simple Storage Service (S3) object storage to save costs, durability or increase scale. Likewise, data may persist to Amazon Elastic Block Store (EBS) if workloads must be transitioned quickly to Amazon Elastic Compute Cloud (EC2) with minimal downtime. It is important in reviewing these solutions that you account for the pluses and minuses associated with your target storage to best fit your migration needs. Target storage options will be discussed in more detail in a following section.

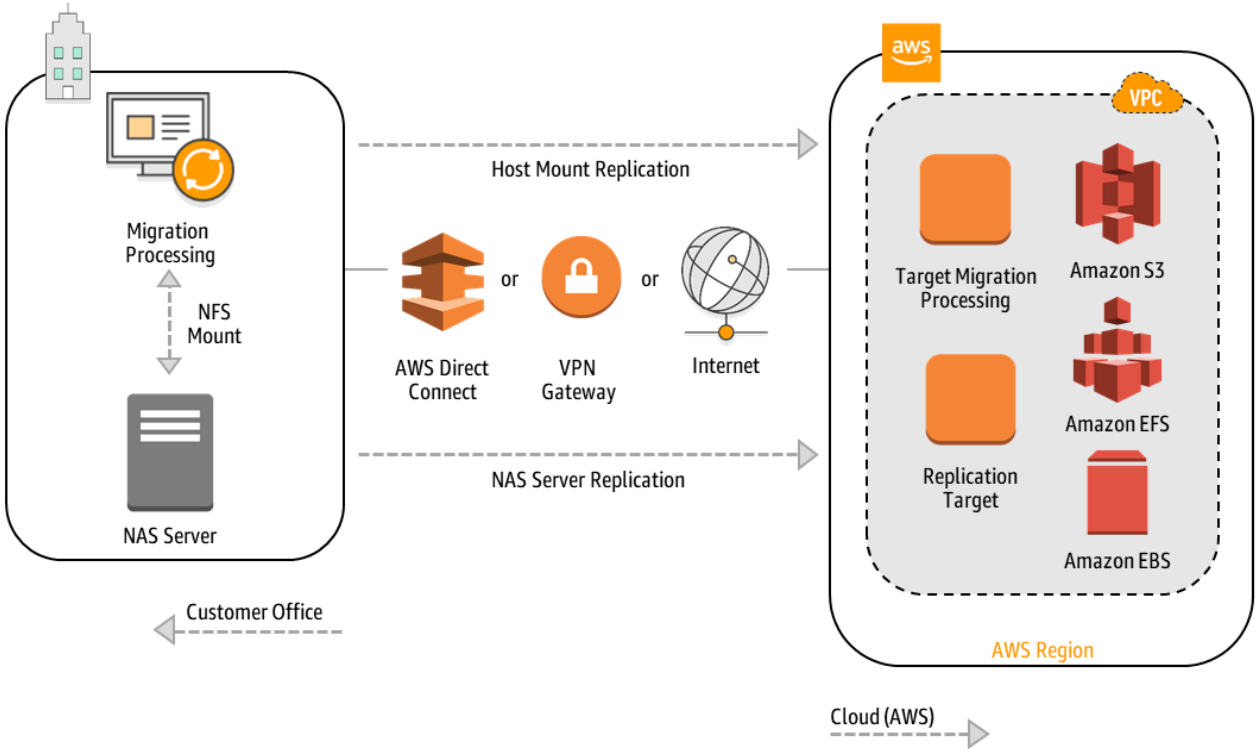
Disaster Recovery strategies and solutions are out of scope for this paper, even though many of the APN partner solutions described here can extend into disaster recovery. Disaster recovery plans still should be maintained after the migration to AWS. AWS offers multiple regions to maintain the geographic business continuity established for mission critical applications. For more information on partner solutions for disaster recovery, visit the AWS Storage Competency page at: https://aws.amazon.com/backup-recovery/partner-solutions/

File Shares

File shares, often called network-attached storage or “NAS,” are a shared repository for often user-generated data. It's common to share file data in organizations across different types of computers, and therefore protocols for both Windows and Unix or Linux are usually supported for the same data set. The most common protocols are Network File System (NFS) and Microsoft Server Message Block (SMB). . NFS has been the standard for Unix platforms since 1984, and is also prevalent in Linux environments. Common Internet File System (CIFS) is based on SMB standards for Windows. Understanding which protocols are used helps determine which tools are appropriate for replicating data to AWS and where it will eventually persist.



In this section, we will discuss two methods of migrating file shares. The first methodology utilizes a host system and software on-premises to mount the file system and replicate it to AWS. The second utilizes NAS vendor-specific replication software to migrate file data residing on hardware at the on-premises on-premises site to an Amazon EC2 instance version of their primary storage software.



Let's look at a few examples:

Primary Storage Migration

NetApp SnapMirror
NetApp is a very common on-premises NAS storage device in the enterprise today. NetApp also has a primary storage version that runs on the AWS Cloud. A NetApp Cloud Volumes ONTAP High Availability (HA) Cluster on AWS can be utilized as a target device for NetApp's native replication technology, SnapMirror.

SnapMirror is snapshot-based replication software that runs on NetApp systems both on-premises and on AWS. For the initial synchronization, a full copy based on a snapshot of the source NetApp volume is copied to AWS to perform a baseline synchronization. As data changes, new snapshots are created, and block level changes to the file system since the previous snapshot are sent to remote site to update the remote volume. The process is then repeated, sending incremental updates to the target until the target system has enough data to initiate the fail over.

Because SnapMirror uses incremental block snapshots, it is very efficient after the initial data copy. Also, if data deduplication and compression are being used on the source volume, the target volumes will retain this optimization.

You can test the data at the remote site any time after the initial synchronization. A read-only copy can be made available, or you can clone a volume to make it writable in your AWS VPC.
The SnapMirror function also is integrated with Cloud Volumes ONTAP's data tiering. With data tiering, a large portion of the replicated data can go to Amazon S3 instead of Amazon EBS. In many cases, NAS storage houses a significant amount of cold data. Migrating the majority to Amazon S3 instead of Amazon EBS will help you "right size" the data footprint and costs.

NOTE: You must have enough Amazon EBS disk to cover your writes plus any hot data that is moved from Amazon S3 once the migration is complete and the workload is running on AWS. Because Amazon EBS does not support multi-attach, capacity must be assigned to each of the NetApp Cloud Volumes ONTAP nodes in an HA pair.

SoftNAS
The SoftNAS approach is to provide data migration for any CIFS or NFS source at the on-premises data center to a SoftNAS instance in AWS utilizing Amazon S3- or Amazon EBS-backed storage pools. During setup, you can choose to enable deduplication and compression to reduce the amount of data being transferred. Using the SoftNAS Cloud instance on VMware at the local site, the SoftNAS instance mounts the existing source file system and migrates the data to a local AWS Snowball appliance or directly to a SoftNAS Cloud instance running on AWS.

If utilizing AWS Snowball, data is mounted by a SoftNAS Cloud instance once the AWS Snowball job is complete and all data has been persisted to Amazon S3. From there, data must be copied to the file system and storage pools created on the SoftNAS Cloud instance on AWS. Although this option requires data to be copied twice, it avoids data copy over the AWS Direct Connect (DX) or Virtual Private Network (VPN) connections.

Use case notes for vendor specific replication: This solution is a fit for customers looking to move their existing NAS solution to SoftNAS on AWS. Both SoftNAS and NetApp can leverage lower cost Amazon S3 as part of the active file system. This can reduce overall costs but should be monitored to ensure there is enough Amazon EBS to serve the hot data at a performance level expected by users and applications.

This solution also fits when you havehas an existing NetApp or SoftNAS storage solution and would like to move the data to AWS while utilizing much the same management, software and capabilities of your on-premises system. Replication of block changes from snapshots can replicate small files more efficiently.

While these solutions provide efficient data replication to AWS, they do not provide orchestration and automation provided by other solutions to get your applications running quickly and efficiently in AWS.

Transparent Tiering Solutions

Komprise
For those looking for a more gradual approach of migrating NAS data to the cloud, Kromprise can extend the archival capability to replicate the entire data set to Amazon S3 and make it available on AWS through CIFS and NFS protocols. The solution starts by installing observers at the on-premises site and connecting these observers to the file system share though NFS or CIFS protocol. Once connected and installed, observers can provide analytics on what data has been accessed and what data would benefit from archive.

Then according to policy, observers replicate metadata information to a director running in an Amazon EC2 instance and persist data directly to Amazon S3. You can set a policy to move all the data from the on-premises NAS system to AWS using the same archive processing. Once the data has been copied to Amazon S3, it can be accessed through both NFS and CIFS. This is done through the Komprise Cloud Controller instance running on AWS

Data Frameworks
ClarityNow from Data Frameworks provides the ability to archive, replicate, or restore an active file system on-premises. Data Frameworks builds a “like” file system structure in Amazon S3 for data that has been migrated to AWS and allows all data to be viewed as a unified file system. You can automate data movement by enabling workflow automation with tagging. With ClarityNow, you can couple the filesystems to key applications and set policies by application priority instead of file age, usage or size.

Use case notes for transparent tiering: Transparent tiering is a good choice when you want to gradually move data to AWS based on usage, or application or data type. Although the primary use case for these technologies is not migration, they provide access to user data throughout the entire movement process, making them less disruptive but typically slower than other solutions for migration.

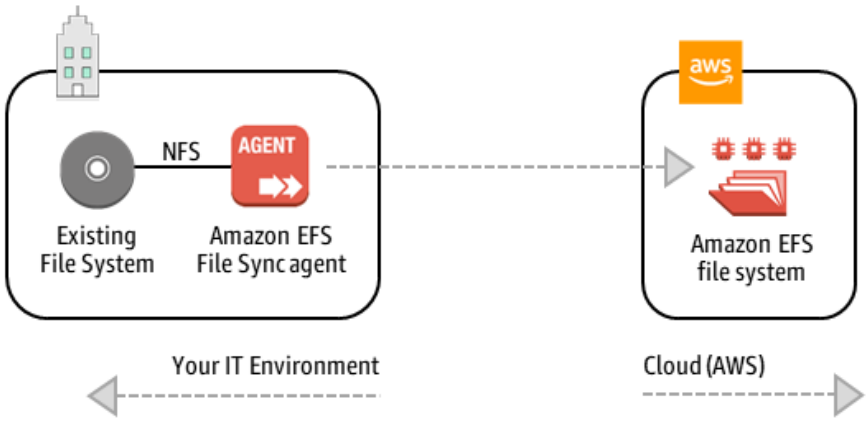
Client Mount and Sync

NetApp Cloud Sync
NetApp Cloud Sync is another software product from NetApp that can migrate your data from on-premises NFS and CIFS to Amazon S3, Amazon EFS or any other CIFS, object or NFS target. In Cloud Sync, you identify the source data on CIFS, NFS or object storage and add a Data Broker instance in your AWS account that controls the synchronization relationship as well as the updates to the data target of your choice in AWS.

Once your Data Broker is deployed, you can choose to synchronize data immediately or schedule synchronization. Once data is synchronized, you can opt to delete the sync relationship. This action does not delete the Data Broker or the data on either the source or the target.
NetApp Cloud Sync is available on the AWS Marketplace here:
https://aws.amazon.com/marketplace/pp/B01LZV5DUJ
Because Cloud Sync is sold as a service though the AWS Marketplace, it can be a very cost efficient way to migrate your data without any long-term software licensing commitment.

Amazon Native Solution: Amazon Elastic File System File Sync
Amazon Elastic File System (EFS) File Sync installs as a virtual machine (VM) on-premises or as an instance in Amazon EC2 in the Amazon VPC. From there, EFS File Sync examines the source and destination NFS file systems to determine which files to sync to Amazon EFS. It does so by recursively scanning the contents of the source and destination file systems for differences. The files it examines include files that have been modified, deleted, added and files that have modified metadata.

Once Amazon EFS File Sync determines the files for synchronization, it copies file data and file system metadata such as ownership, timestamps, and access permissions to the Amazon EFS target.



The amount of time Amazon EFS File Sync spends in the preparing status depends on the number of files in both the source and destination file systems, as well as the performance of the file system. When a sync task starts, Amazon EFS File Sync performs a recursive directory listing to discover all

files and file metadata in the source and destination file system to determine the files to sync. This will take longer as more files are present on source and target file systems. Once your data sync is complete, delete the sync agent you created and remove the VM or instance if it is no longer needed.

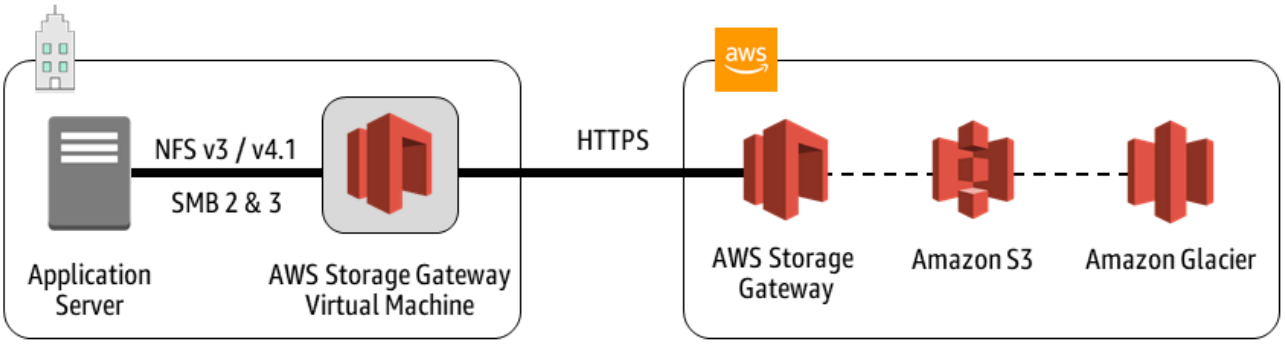
Use case notes for Client Mount and Sync: Client mount and sync provides a more universal approach to data movement because the file share is mounted directly from a client or proxy running the migration software. File system scans scan significantly affect performance is the file system is composed of many small files. Also with this methodology, CIFS or NFS must be chosen to mount the data. If the dataset is used by both CIFS and NFS clients, you may want to consider primary storage replication as a first option.

AWS Storage Gateway Solution Example

AWS Storage Gateway can migrate data to AWS in three ways: files, volumes and virtual tapes (VTL). you can install Storage Gateway on a VMware instance running on-premises or in an Amazon EC2 instance in your Amazon VPC. For purposes of this section, we will focus on files.

With file gateway, files are stored as objects in your Amazon S3 buckets and you can configure the initial storage class for objects that file gateway creates. There is a one-to-one relationship between files and objects, and you can configure the initial storage class for objects that file gateway creates. The object key is derived from the file path within the file system. For example, if you have a gateway with hostname *file.amazon.com* and have mapped *my-bucket*, then file gateway will expose a mount point called *file.amazon.com:/export/my-bucket*. If you then mount this locally on */mnt/my-bucket* and create a file named *file.html* in a directory */mnt/my-bucket/dir* this file will be stored as an object in the bucket *my-bucket* with a key of *dir/file.html*.

File gateway supports Linux clients connecting to the gateway using NFS versions 3 and 4.1 for Linux clients, and supports Windows clients connecting to the gateway using SMB versions 2 and 3. For the backend storage, file gateway supports Amazon S3, Amazon S3 Standard - Infrequent Access (–Standard-IA) and Amazon S3 One Zone - IA. You can move data to Amazon Glacier utilizing life cycle policies but you will receive a generic I/O error if the user or application tries to retrieve that file.



Use case notes: Data needs to be moved to AWS Storage Gateway to enable access to the file data as it resides in Amazon S3. This use case is great when you want to migrate file data to Amazon S3 but still want legacy access on-premises through NFS or SMB. Storage Gateway also can be instantiated in the Amazon VPC to provide test and development access to clones of the primary data set.

Application insanitation and recovery automation is outside the scope of this solution as it is very focused on data movement at the disk or file level.

Unmanaged Tools
Tools such as rsync can be used to move data from file shares as well as local block storage. Rsync can copy locally, to/from another host over any remote shell, or to/from a remote rsync daemon. Rsync reduces the amount of data sent over the network by sending only the differences between the source files and the existing files in the destination and is widely used for backups and mirroring and as an improved copy command. Although there are variants that would also extend rsync to Amazon S3, it is recommended that you utilize the sync parameter in the Amazon S3 command line interface (CLI).

The Amazon S3 CLI allows you to utilize your own scripting to copy and store data from on-premises servers to Amazon S3. The Amazon Glacier CLI targets Glacier vaults for migrating data for very infrequently accessed archive data.

Use case notes: This is a good option for low-cost migration of limited data sets to AWS. In most cases, it is very difficult and ineffective to utilize these tools for larger data sets where deduplication and compression can reduce the data both sent to cloud and stored in cloud.

Best Practices for File Migration

It is always more efficient to transfer data with multiple Transmission Control Protocol (TCP) streams. Having multiple streams will improve overall throughput in almost every case. When preparing for data synchronization, utilize different mount points and file system mount points to create multiple sync jobs.

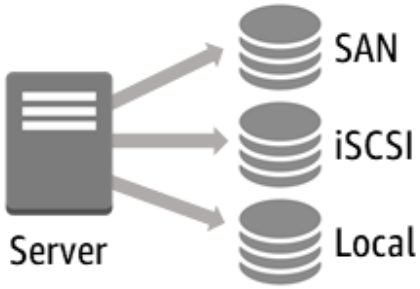
Consolidate/archive small files whenever possible if using replication technology which scans the file system. The file scanning processes to synchronize changes can be costly. The more files that are present in a file system will have a direct effect on the speed of synchronization processes. In cases where there are many small files, consider replicating with block level technology such as NetApp SnapMirror or CloudEndure.

Finally, when utilizing a proxy method, try to use read-only snapshot from the source. This will ensure that files at the source location can’t be modified while the files are being transferred, and makes sure that verification works.

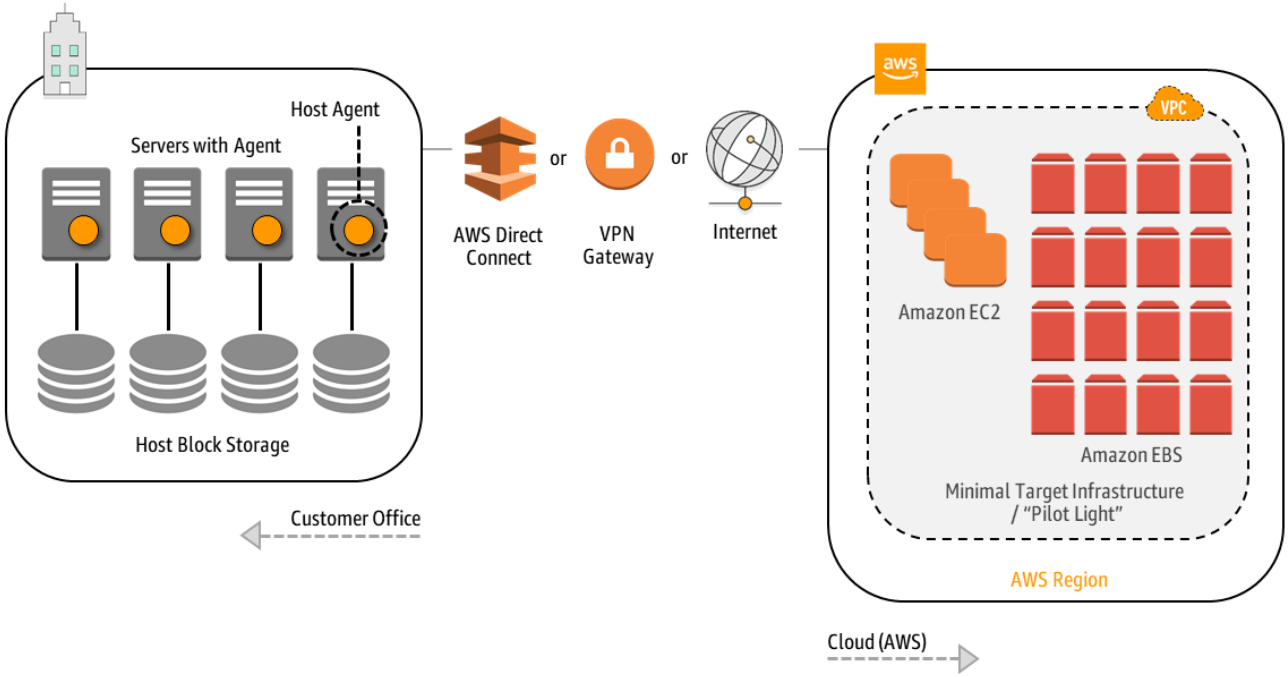
Host-Attached Block Storage

Host block storage is accessed from the host operating system, typically through a local file system. The volumes are typically provisioned to the host three ways: Storage Area Network (SAN), Internet

Small Computer Systems Interface (iSCSI) and local disk. This paper will combine the use case for these different interconnect options under host block storage.



Migration of host block storage covers a very wide group of servers and applications targeted for cloud migration. There are two basic components for replicating host block storage. The first is host agent software, which intercepts and buffers the block write to a host's block storage and then replicates that block to the remote site. Replicating at the block level provides a crash-consistent copy of the data at the remote site without the need for specialized application integration or complex scripts. It is important to ensure your application can recover from a crash-consistent copy. Failover testing is a must for this methodology.



Host Agent Block Replication

CloudEndure

The CloudEndure agent operates as an I/O filter driver and sees data changes on block devices in real-time. It provides continuous data protection by replicating all new and changed blocks on the physical or virtual source hosts to the target site. Since the agent is tightly integrated with the operating system, be sure verify compatibility and support:

https://docs.cloudendure.com/Content/Getting_Started_with_CloudEndure/Supported_Operating_Systems/Supported_Operating_Systems.htm

Unlike a snapshot or scheduled synchronization, continuous replication covers both baseline data and ongoing changes. On the target side, a skeleton infrastructure of automatically deployed and managed servers receive updates and persist changes to Amazon EBS volumes. The skeleton infrastructure combines volume updates on as few servers as possible to keep infrastructure minimal during the migration. Once replication "lag" is zero, meaning all data has been replicated to the target, a new option is presented. In order to ensure the migrated workload will run as expected in AWS, CloudEndure analyzes performance data collected on the source. When you invoke the cutover operation in CloudEndure, it re-provisions infrastructure based on the needs of the server and volume. If you decide the cutover operation was a test, you can revert to the skeleton infrastructure and catch up from the original target again. If you decide the cutover operation was a real cutover, you can leave your workload running in AWS and suspend on-prem operations.

Unlike the virtual disk migration use case, you are required to install an agent into every VM or physical server using this method of migration. Although this adds additional host overhead and more installation management, this use case can provide a more expedited failover to AWS and consume less resources for testing the cloud migration of workloads and applications.

Commvault ContinuousDataReplicator
If you already utilize Commvault for your backup infrastructure, you can add ContinousDataReplicator (CDR) to migrate data to AWS. CDR allows near-time continuous data replication for critical workloads. Replication can be configured as direct replication (1:1 source to destination host), or as a fan-in or fan-out based replication configuration. CDR performs asynchronous block based replication of the host block storage.

The CDR host software is integrated with snapshot capabilities on the local host to also provide application-consistent updates, including SQL and Exchange, to be sent to the remote site. This also extends to NetApp ONTAP snapshots. In this case, if the hosts has mounted an NFS, CIFS, or iSCSI volume, CDR can send the change blocks from those snapshots to the remote site.

There are a lot of factors that affect scalability of the fan-in or fan-out configuration including number of changes, disk space available for logging, and network constraints. You can find details of these limits here:
https://documentation.commvault.com/commvault/v10/article?p=features/data_replication/fan_in.htm

For fan-in, you must keep the ratio less than 100:1 to follow best practice. For fan-out, storage for the replication journal is required for each source and destination host. You must ensure that enough disk space is available to support each replication pair.

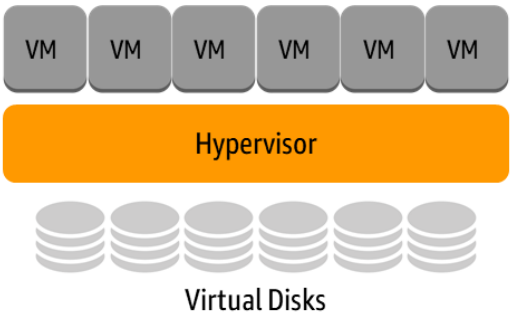
Use case notes: Replication is independent of application (provided application can recover from a crash consistent copy). This also avoids additional application licensing on the remote target in many cases. Since a host agent is installed on each host, you must be mindful of operating system support including patches and upgrades. RPO and RTO are closer to zero with continuous replication rather

than snapshot based replication methodologies.

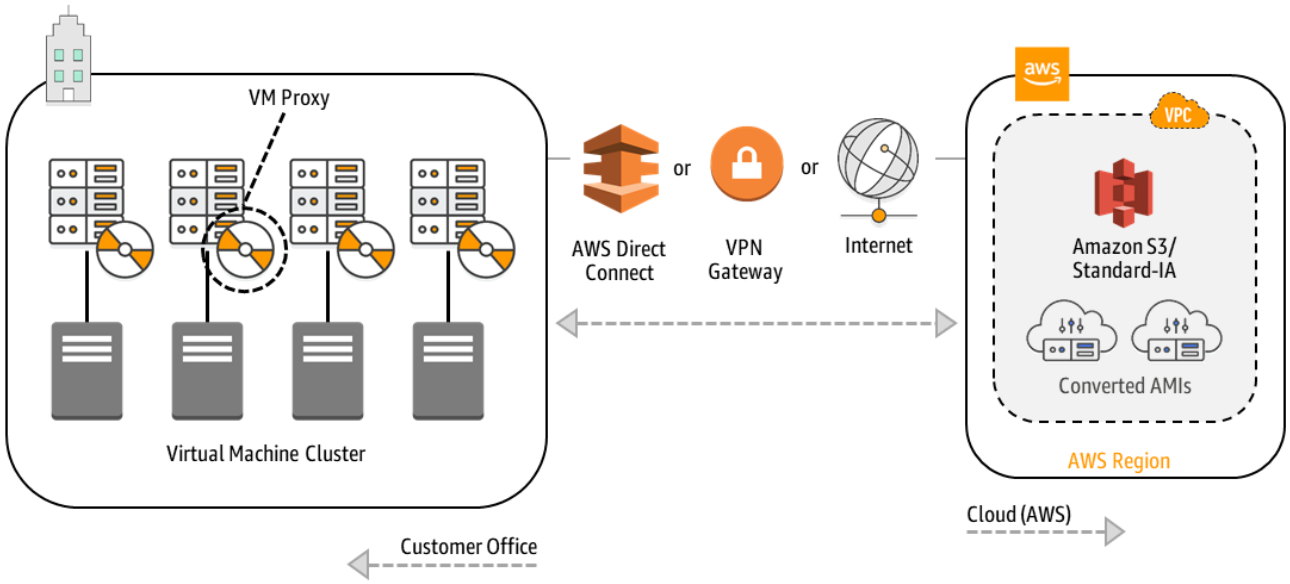
Unmanaged Tools
For more information on Unmanaged Tools, please refer to the Client Mount and Sync section of this paper.

Virtual Disks

With virtual disk storage, volumes are presented to the virtual machines through the hypervisor. From there they appear as local block storage to the VM operating system. Migration for this type of disk, then, can utilize the options for host block storage or VM-specific migration capabilities enabled with proxy servers and hypervisor specific APIs.



Migrating VMware and Microsoft Hyper-V data to AWS utilizes the built-in APIs and primitives for backup and migration for virtual environments. Here a VM acts as a proxy to synchronize the disk images from a VM cluster. This eliminates the need for each VM to have host agents or clients running in the guest operating system. All disk, file indexing, and change block tracking is received from the APIs of the virtual infrastructure.



AWS Server Migration Service (SMS) installs a proxy server in the VM cluster to migrate VMs to AWS. AWS SMS provides the scheduling to replicate volumes of live VM servers to AWS and the creation of new Amazon Machine Images (AMI) periodically while tracking progress. You can schedule initial replications, configure replication intervals, and track progress for each VM using the AWS console.

Partner solutions employ a similar VM cluster proxy methodology and support additional use cases such as backup and recovery, disaster recovery, failback/test, and on-going migration. Here are a few examples:

Veritas Resiliency Platform
Orchestration for migration operations as well as pre-migration testing are features of Veritas Resiliency Platform (VRP) allow users to test how applications will perform in AWS before the move.
Solution components are made up of on-premises virtual machine images available on Veritas' website and AWS cloud components available in the AWS Marketplace.

For the AWS-side deployment, a Resiliency Manager Instance is the control plane and point of user interaction. The Infrastructure Management Server instance does the job of discovery and operations during migrate and DR workflows – provisioning and de-provisioning Amazon EC2 instances. Instance provisioning is done at the time of configuring workloads for migration.

The Replication Gateway Instance interacts with Amazon S3 bucket and prepares the target Amazon EBS volumes. The Replication Gateway also pushes changed blocks (update sets) to Amazon S3 during reverse path.

The cloud native infrastructure of VRP utilizes AWS Lambda for processing Amazon S3 notifications as well as for processing API requests coming via API gateway. Amazon Simple Queue Service (SQS), Amazon Simple Notification Service (SNS), and Amazon Dynamo DB also are used

to ensure ordering and purging of Amazon S3 objects. And finally, the data gateway is used for replication traffic in both directions.

For an on-premises deployment, Infrastructure Management Server does the job of discovery and orchestration during migrate and DR workflows. The Replication Gateway Instance interacts with the Amazon S3 bucket and prepares the target virtual machine disk/virtual hard disk volumes or pushes changed blocks (update sets) to Amazon S3.

VRP use cases include migrate, takeover/failover, rehearsing, starting, stopping, and you can implement custom actions like running scripts pre and post migration.

Zerto
Zerto has a unique approach to replicating data for virtual disks. Zerto installs a VM on each hypervisor and captures, clones, and sends every write from a specified group of VMs virtual disks. This VM is called the Zerto Replication Appliance (ZRA). Updates are then sent to the Zerto Cloud Appliance (ZCA) instance running on AWS and updates are persisted to Amazon S3.

The Zerto Virtual Manager (ZVM) is the central management interface for replication and recovery orchestration, deployed in a Windows VM. You must deploy one ZVM per VMware Virtual Center or Systems Center Virtual Machine Manager (SCVMM).

Commvault LiveSync
Commvault LiveSync is a good option when you either have Commvault Commcell infrastructure in place or are considering adding one. Much like on-premises VMware and Hyper-V backups, Commvault LiveSync uses a proxy on VMware and Hyper-V to replicate the block changes of VM images. Once the VM image is in AWS, Commvault orchestrates the conversion, updates, failover and failback of replicated VMs. Failover options include:

- **Test Boot:** this scenario takes a snapshot of the VM before the test boot, boots destination VMs with network connections disabled, and reverts to the snapshot afterwards.

- **Planned Failover:** This option powers off the source VMs.and performs an incremental backup of the source VMs to capture the latest data. Once data is applied to the destination VMs, remote instances can be powered on in AWS.

- **Unplanned Failover:** this option would not be used for a migration but should be noted for on-going disaster recovery requirements.

As you can see, the migration process includes VM downtime to synchronize the latest updates to complete the migration. In practice, you should have a good understanding of the time incremental backups take to complete and plan your migration cutover accordingly. Incremental job duration can be obtained from the Commvault Commcell console.

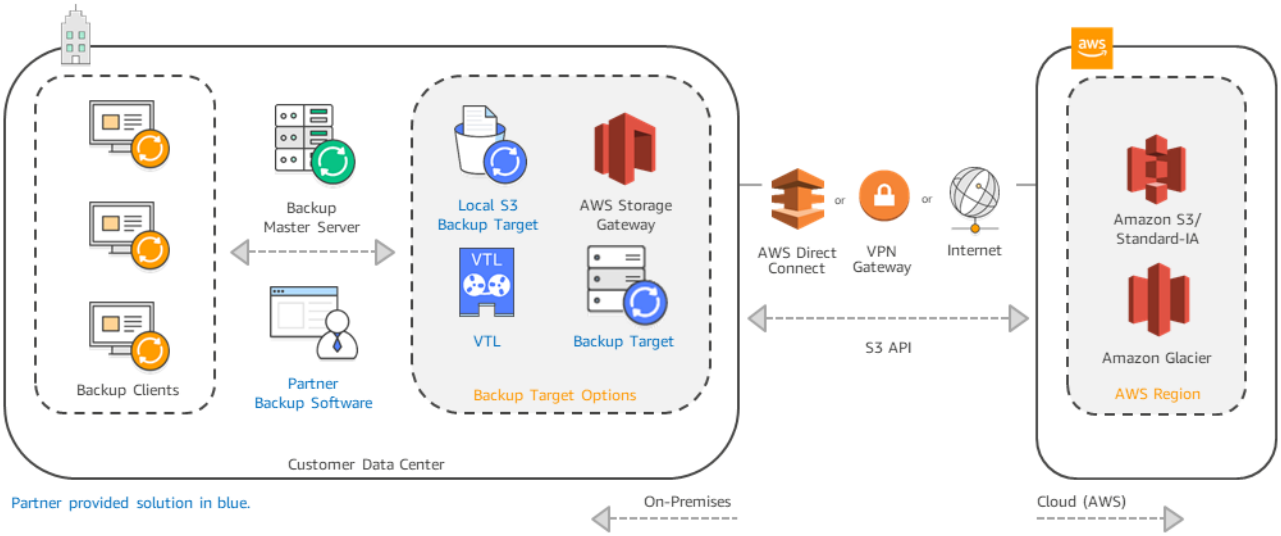
Amazon Server Migration Service
AWS Server Migration Service (SMS) installs a proxy server in the VM cluster to migrate VMs to AWS. AWS SMS provides the orchestration to replicate volumes of live VM servers to AWS and

the creation of new AMIs periodically. Orchestration includes scheduling replications and tracking progress for a group of VM servers. Options include scheduling of initial replications, configuring replication intervals, and tracking progress for each server using the AWS console.

Because the proxy runs in the existing VM cluster, this method simplifies configuration and installation to an existing VM cluster and provides a migration that minimizes downtime during the data movement and conversion process.

Backup and Recovery

Utilizing the existing backup and recovery solution at the on-premises site can be a very cost effective an efficient way to migrate data to AWS. AWS storage partners provide compelling solutions to replace traditional backup, restore, and archive environments. These cloud-enabled backup solutions provide ease of use, lower costs, and scalability by leveraging Amazon S3, Amazon S3 Standard-IA, AWS Storage Gateway, and Amazon Glacier as cloud storage targets.



Options to connect your backup environment to AWS include:

AWS Storage Gateway

If the backup solution lacks a native connector to AWS storage, you can still utilize AWS Storage Gateway in VTL mode. Once data is in cloud, it can be restored to instances from the backup vendor software and AWS Storage Gateway running in the customer Amazon VPC.

Backup Vendor Connectors

Here the existing backup solution can target full and incremental backups directly to Amazon S3 and Amazon Glacier though native connectors. Most APN Technology Partners with this integration also have built-in d-duplication, encryption, and compression to reduce both storage capacity on AWS and the amount of data sent across the network. Connectors also support the scenario where backups are still kept locally on disk or tape and copied to AWS immediately or by policy,ensuring that backup windows do not increase during the movement of data to cloud.

Appliance with tiering to Amazon S3
In addition to AWS Storage Gateway, there are partner appliances which provide a target for backup data on-premises and automatically tier data by policy to AWS. Depending on the appliance solution, the backup software may be included, or the appliance may work in conjunction with a wide range of backup software vendors. These appliances often include deduplication, compression and encryption options.

Use case notes: Utilizing your existing backup and recovery solution can leverage existing infrastructure, target specific AWS storage classes, and apply data reduction, making it a very cost effective option for migrating data. In most cases, it is not appropriate for applications that need very minimal disruption during migration. Although the data movement processing is non-disruptive, there is restore, compatibility, and potential re-factoring of applications to utilize the backup data in AWS. For more information on APN backup and restore partners, reference this whitepaper here.

AWS Snowball and AWS Snowball Edge

AWS Snowball is ideal for transferring large amounts of data, up to many petabytes, in and out of the AWS cloud securely. This approach is most relevant when you do not wish to make expensive upgrades to your network infrastructure just to get a large amount of data migrated. In general, if migrating your data using would take a week or more, you should consider using AWS Snowball or AWS Snowball Edge.

There are a couple of key difference between AWS Snowball and AWS Snowball Edge. AWS Snowball Edge has the capacity for local compute as well as a file interface with NFS support. You also can run AWS Snowball Edge in cluster mode, where multiple AWS Snowball Edge devices can act as a single, scalable storage and compute pool with increased durability. AWS Snowball Edge also has the ability to run AWS Lambda functions as data is copied to the device.

With both versions of AWS Snowball, AWS transfers your data directly onto and off of AWS Snowball storage devices using our high-speed internal network, bypassing the Internet. For data sets of significant size, AWS Snowball is often faster than Internet transfer is and more cost-effective than upgrading your connectivity. Once your data is migrated to AWS, data can be copied or moved to other AWS services such as Amazon S3 and Amazon Glacier for longer term retention.

For backup and recovery solutions, AWS Snowball has wide (but not complete) support of many backup software partner solutions for off-line synchronization of a full baseline backup. This can provide a much more predictable method for moving large amounts of data to AWS.

Additionally, AWS Snowball can be used with a few of the partner solution described in this document for initial synchronization. The use of Snowball typically requires multiple manual steps to setup. It is recommended you view the partner documentation for these steps before ordering your Snowball through the AWS console. Here are some helpful links describing the AWS Snowball setup for these solutions:

- Commvault: https://documentation.commvault.com/commvault/v11_sp6/article?p=features/cloud_storage/t_seeding_amazon_S3_snowball.htm

- SoftNAS: https://docs.softnas.com/pages/viewpage.action?pageId=6520834

Storage Migration Targets in AWS

The next step in planning is determining the data migration target in AWS. Here is a brief overview of the storage services and solutions used as targets in this paper.

AWS Native Storage Services

Amazon EBS

Amazon EBS presents persistent block storage volumes for use with Amazon EC2 instances in the AWS Cloud. Each Amazon EBS volume is automatically replicated to offer high availability and durability. Amazon EBS can deliver performance for workloads that require the lowest-latency access to data from a single Amazon EC2 instance.

Migration notes for Amazon EBS: Amazon EBS should be used as a target when you need fast access to the data in Amazon EC2 once migration is complete, e.g.applications or databases which require minimal disruption during the migration process.

Amazon EFS

Amazon EFS is a file storage service for use with Amazon EC2. Amazon EFS provides a file system interface, file system access semantics (such as strong consistency and file locking), and concurrently-accessible storage for up to thousands of Amazon EC2 instances. Amazon EFS is designed to be highly available and durable. Each Amazon EFS file system object (e.g. directory, file, and link) is redundantly stored across multiple Availability Zones (AZs) within an AWS region.

Migration notes for Amazon EFS: Amazon EFS is a great choice for data that must be shared by many instances or services in AWS. It scales for capacity, support very large file sizes, and provides provisioned performance based on use case need. At the time of this writing, protocol support is

NFSv4, so it is important to ensure applications already support NFSv4 or are re-factored to connect to the shared storage.

Amazon S3

Amazon S3 is an object storage service. Amazon S3 makes data available through an Internet API that can be accessed anywhere and deliver 99.999999999% of durability. Data in Amazon S3 Standard, S3 Standard-IA, and Amazon Glacier storage classes is automatically distributed across a minimum of three physical Availability Zones that are typically miles apart within an AWS Region. The Amazon S3 One Zone-IA storage class stores data in a single AZ, and is ideal for customers who want a lower cost option for infrequently accessed data and do not require the durability and resilience of Amazon S3 Standard storage.

Migration notes for Amazon S3: Amazon S3 provides the lowest cost and also the most durable target for migration data. This makes Amazon S3 the most common target for backup and restore and data migration solutions. When using Amazon S3 as a target, you should consider if the migrated data will stay persisted in Amazon S3 or be restored to another storage service for use by the AWS-hosted workload. For workloads which require more immediate installation or transaction performance, migration to another storage option is recommended.

Data Transfer Considerations

After we have identified the storage data to be replicated and the target storage on AWS, network transport must be assessed. To start, run a quick analysis on your current network bandwidth to see if it is sufficient to migrate the data in a reasonable amount of time. To calculate this, you can use the following formula:

Number of MBs per day = (Megabits per second/8) *

Network Utilization * 60 * 60 * 24

For example, let say you have a 300Mb connection to the internet and can achieve 60% utilization to the Amazon VPC, then you could transfer:

Number of MBs per day = (300/8) *

0.6 * 60 * 60 * 24 = 1,944,000 MB/day = 1.9 TB/day

Common Issues that Affect Network Throughput

If you are not achieving your expected throughput during a storage migration, there are a few common bottlenecks to check.

Many small files to transfer – The performance of any NFS copy will decrease significantly if there is many small files in the file system to be migrated. The reasons are the increased read write operations and network overhead associated with handling more files in the copy process. If you have a file system with millions of small files, you can try to mitigate the performance impact by utilizing multiple streams and consolidating files to larger archives where appropriate. Even with these steps, you should plan on slower than expected performance for this type of data profile.

Network traffic shaping – Network administrators may be throttling bandwidth utilized by what would be identified as a backup process. This can significantly reduce effective bandwidth especially during certain times of the day. It is important to communicate with your network team and understand what your effective bandwidth will be throughout the migration process.

Latency – Latency can significantly affect overall bandwidth to the migration target. Closely monitor your latency during different periods of the day to assess the potential impact.

Utilizing internet bandwidth can produce inconsistent results for data transfer to AWS. This includes but is not limited to latency and hops, limits of firewalls, peering point limitations and DNS lookup issues. To ensure that you can make your migration window, all components of the on-premises network, the transport network and the AWS network need to be considered.

AWS Direct Connect

AWS Direct Connect (DX) provides a high bandwidth backbone that can be used to transfer data between your corporate network and AWS securely without ever having the data routed over the Internet.

With AWS DX, you can reduce network costs, increase bandwidth throughput, and provide a more consistent network experience for your migration than other Internet-based connections.

You can establish a dedicated 1GB or 10GB network connection from the on-premises data center to AWS and use the connection to access your Amazon VPC as well as AWS services, such as Amazon S3.

AWS DX requires a longer term commitment than a migration may require, especially in the case where the existing data center will be retired. In addition to bandwidth requirements, the duration of the migration project should be considered when assessing whether AWS DX should be deployed.

Amazon S3 Transfer Acceleration

Amazon S3 Transfer Acceleration (S3-XA) enables fast, easy, and secure transfers of files over long distances between your client and your Amazon S3 bucket. Transfer Acceleration leverages the Amazon CloudFront globally distributed AWS edge locations. As data arrives at an AWS edge location, data is routed over the AWS backbone to the Amazon S3 bucket for optimal data transfer.

Transfer Acceleration helps you ensure consistently fast data transfer to Amazon S3 minimizing the effect of distance on throughput. Generally, you will see more acceleration when the source is farther from the destination, when there is more available bandwidth, and/or when the object size is bigger. AWS provides an online speed comparison tool so you can assess the performance benefit from uploading data through Amazon S3 Transfer Acceleration.

Enabling Transfer Acceleration on a bucket can be done with a single click of a button in the Amazon S3 console; this makes the accelerate endpoint available to use in place of the regular Amazon S3 endpoint.

Finally, you pay only for what you use and for transferring data over the accelerated endpoint. Transfer Acceleration has the following pricing components: data transfer in (per GB), data transfer out (per GB), and data transfer between Amazon S3 and another AWS Region (per GB). Transfer acceleration pricing is in addition to data transfer (per GB per month) pricing for Amazon S3.

Conclusion

Choosing the right method for migrating storage to AWS depends greatly on the category of storage targeted for migration. Three storage types which encompass the most common storage use cases in the enterprise today are NAS file shares, host-attached block storage, and virtual disks. While choosing either a native AWS solution, a partner solution, or a combination of both, migration plans should consider factors such as cost, efficiency, durability and application workload requirements. This paper outlines those vectors for each solution in the context of the three most common use cases in data storage migration today.

Document Revisions

Date	Change	In sections
October 9, 2018	Initial publication	—

Additional Documentation & Resources

AWS SMS: https://docs.aws.amazon.com/server-migration-service/latest/userguide/server-migration.html
Amazon EFS File Sync: https://docs.aws.amazon.com/efs/latest/ug/walkthrough-file-sync-onpremise.html#create-sync-task
AWS Storage Partner Competency: https://aws.amazon.com/backup-recovery/partner-solutions/