EMC ECS Architectural Guide v2.x

High Availability/Disaster Recovery/Reliability in a Multi-site Scenario

ABSTRACT

This paper on EMC ECS (Elastic Cloud Storage) provides an in-depth discussion that includes hardware and software components, networking, and architecture of the hyperscale, software-defined and geo-distributed cloud storage platform. This white paper can be an educational as well as a reference document.

June, 2015

To learn more about how EMC products, services, and solutions can help solve your business and IT challenges, contact your local representative or authorized reseller, visit www.emc.com, or explore and compare products in the EMC Store

Part Number H14071.1

Table of Contents

INTRODUCTION

EMC ECS (Elastic Cloud Storage) provides a complete software-defined cloud storage platform for commodity infrastructure. Deployed as a software-only solution or as a turnkey appliance, ECS offers all the cost advantages of commodity infrastructure with enterprise reliability, availability and serviceability. Built for hyperscale and Big Data applications, ECS provides support for both object and HDFS. Its scale-out and geo-distributed architecture enables customers to have an in-house cloud platform that scales to exabytes of data with a TCO (Total Cost of Ownership) that's up to 28% less than public cloud storage services.

This paper gives detailed information on the hardware components, the internals of software, networking architecture, and high-availability features such as geo-federation and geo-replication.

1 ECS FUNDAMENTALS

In the next few sub-sections, we will cover the fundamentals of ECS hardware and software architecture.

1.1 ECS Hardware Architecture

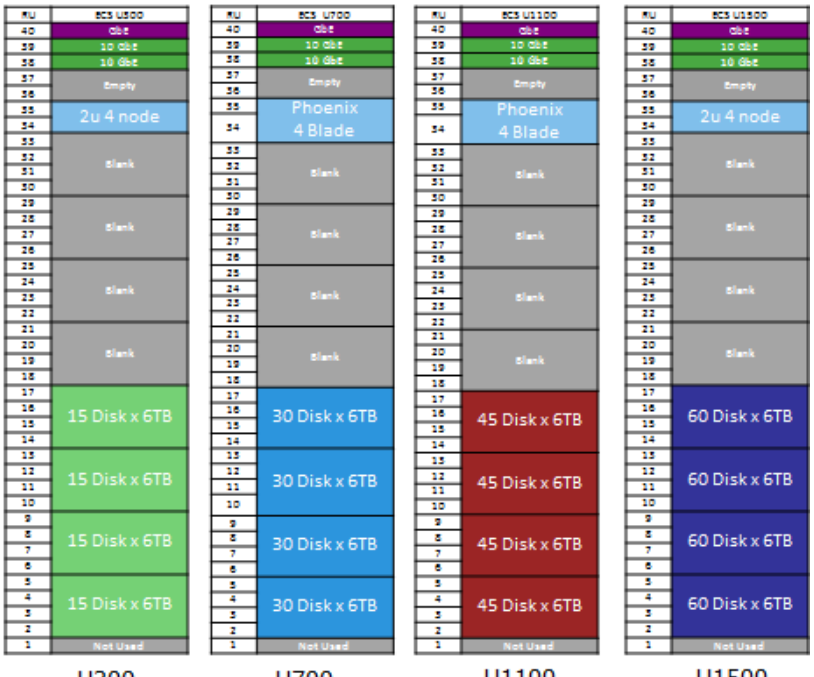
ECS is a hyperscale cloud-storage platform that is a combination of intelligent software and commodity hardware. Its components are widely available, inexpensive, standardized, interchangeable, and compatible with other similar components. This enables ECS to provide a turnkey appliance with a scale-out architecture of high performance at relatively low cost.

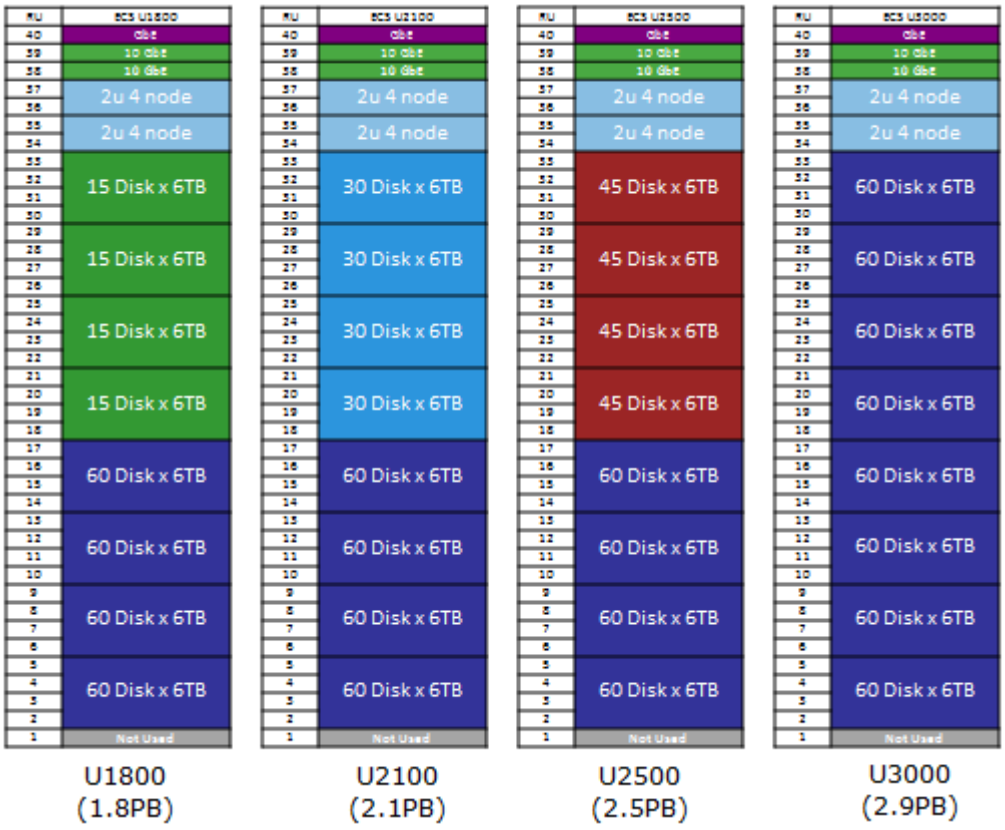
ECS ensures broad application support – object and HDFS today, with native file support planned for the future. It can be configured as a single or multi-site distributed cloud topology. When ECS Appliance is configured as multi-site distributed cloud topology, ECS Appliance provides geo-replication and multi-site access as single name space. It supports popular APIs including Amazon S3, Atmos REST, Centera CAS and OpenStack Swift.

1.1.1 ECS Appliance Configuration

There are eight standard models for ECS Appliance that range from 360TB to 2.9PB by the number of the storage; there are also optional "C" models that have fewer drives and higher server-to-disk ratio.

Each standard model contains two 10GbE and one 1GbE switches and four or eight x86 server and disc enclosures. Each disk enclosure is connected to one x86 Server by SAS; a four-server model has four disk enclosures and an eight-server model has 8 disk enclosures. A disk enclosure can have up to 60 disks.





The following is a list of various ECS Appliance models.

Model	U300	U700	U1100	U1500	U1800	U2100	U2500	U3000
Nodes	4	4	4	4	8	8	8	8
Disks	60	120	180	240	300	360	420	480
Raw TB	360	720	1,080	1,440	1,800	2,160	2,520	2,880
Max Power kVA	3.6	4.1	4.6	5.1	8.1	8.6	9.1	9.6
Max Heat BTU/h	12,150	13,838	15,525	17,213	27,338	29,025	30,713	32,400
Weight lbs.	970	1150	1,330	1,510	2,020	2,200	2,380	2,560

The C-models are basically ECS nodes without the DAEs – the nodes just have drives that fit in the internal drive slots. Three drives for data per node are used. Fewer drives in these ECS models equates to a very high server-to-disk ratio which will be suitable for applications with high performance needs. The drives are, right now, 6TB HDD. More details on these appliances will be coming soon. The C70 models are sold with a minimum of 8 nodes -- 2 Phoenix blade servers -- which equate to 24 disks and a raw capacity of 144TB (24 x 6TB).

Note that a single ECS system can be scaled beyond eight nodes by daisy-chaining the 1Gb switches (for management traffic) and also linking the 10Gb switches in the racks to a customer-provided switch/backplane (for data and chunk traffic). Technically, such a multi-node system will be called a VDC -- Virtual Data Center. When the 1Gb switches are interlinked, they use a vLan called Nile Area Network" or NAN.

1.1.2 Hardware Components

An ECS Appliance rack includes nodes, disk enclosures, disk drives, 10GbE switches and a 1GbE switch.

ECS Appliance Server

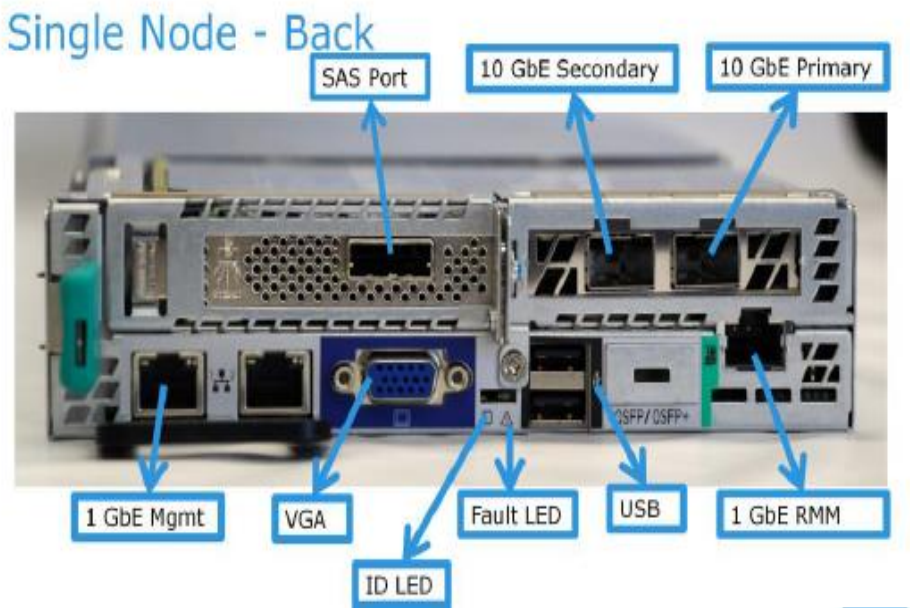
ECS comes with 4-blade servers, each of which has four built-in "pizza-box" servers in a 2U chassis. Server specs are as follows.

Phoenix 4 blade

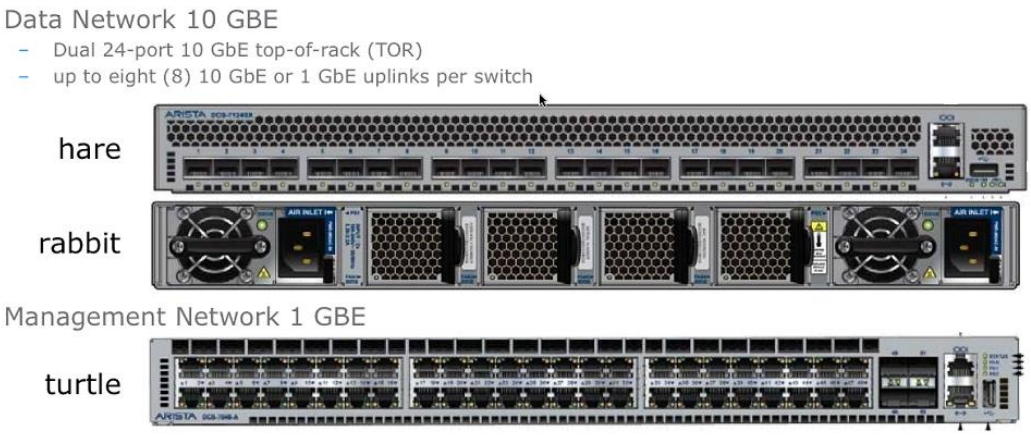
- 4 Server node in 2U
- Dual 4-core Intel 2.4GHz IvyBridge (E5-2609)

(will be upgraded to Haswell CPUs in 2015)

- 64GB memory per node
- Dual 10Gb Ethernet (NIC-bonding makes them appear as a
- Dual 1Gb Ethernet
- Dedicated 1Gb Ethernet remote diagnostics
- 4 x 2.5" drive slots per node
- Single DOM for OS boot
- Provide Disk Attachment point



ECS Appliance Switches



There are two 10GbE switches and one 1GbE switch within the ECS appliance rack. The 24-port 10GbE switches -- Arista 7150S-24 -- are used for data transfer to and from customer's applications. These switches are connected to the ECS nodes in the same rack. The internal names for these two 10Gb switches are Hare (the top one) and Rabbit (the one below). These two switches are linked through MLAG (Multi-Chassis Link Aggregation) and thus appear as one large switch to the customer's network or application.

The 48-port 1Gb Arista switch -- known within EMC as "Turtle" for its slower speed -- is used by the ECS nodes for

a. internal communication of data and management between the nodes, and

b. out-of-band management communication between the customer's network and the RMM ports of the individual nodes.

More details of the switch configurations are found in Sections 2.1.2 & 2.1.3 of this white paper.

ECS Appliance Disk Enclosure

DAEs are 4U enclosures that can have a maximum of 60 drives arranged in 5 rows and 12 columns as shown in the diagram *(to the right)*. Each DAE is connected to one server in the ECS node by one SAS connection. Thus, there will be four DAEs in a 4-node ECS rack, and eight DAEs in an 8-node ECS rack.

Currently, the supported drives are 1TB and 6TB, both being 7200 RPM SATA drives. Each DAE can have 15, 30, 45 or 60 drives depending on the ECS model. Drives of higher capacity – 8TB and 10TB – are expected to be supported later in 2015 and early 2016.

Each 60-drive DAE includes a hot-swappable LCC. The LCC's main function is to be a SAS expander and provide enclosure services for all 60 drive slots. The LCC independently monitors the environmental status of the entire enclosure and communicates the status to the ECS Element Manager. Note that the LCC is a hot-swappable FRU but it cannot be replaced non-disruptively. Since there is only one LCC in the DAE, the disks in the DAE will need to be brought offline before the LCC is replaced.

As for the DAEs, they don't have built-in data protection from a software RAID or a hardware RAID controller. However, ECS software has other ways to protect the data, which will be discussed later in this paper.

ECS Node Connectivity

There are four servers or ECS nodes in a 2U blade server, and each server (ECS node) has one SAS connection to a DAE, and has four network connections -- one link to each of the two 10Gb switches and two links to the 1Gb switch.

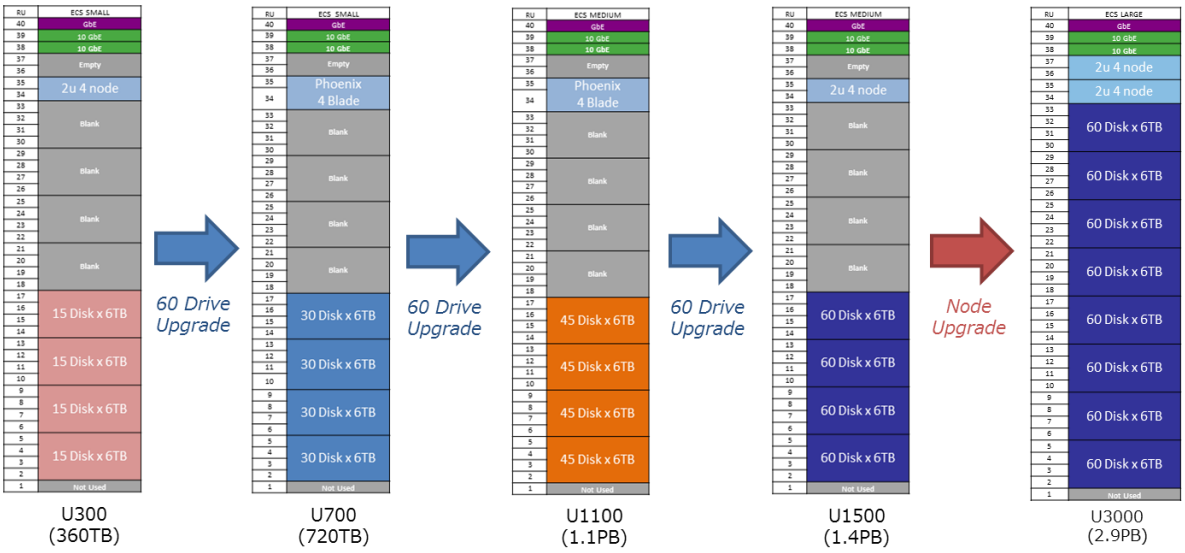
Each ECS node has two 10Gb ports which appear to the outside world as one port, thanks to NIC bonding. Each of these 10Gb ports is connected to one port in each of the 10Gb switches (Hare and Rabbit). These public ports of the ECS nodes typically get their IP addresses from the customer's DHCP server.

The 1Gb management port in an ECS node is connected to an appropriate port in the 1Gb switch (Turtle) and has a private address of 192.168.219.X. Each node also has a connection between its RMM port and a port in the 1Gb switch which in turn has access to a customer's network to provide out-of-band management of the servers. To enable access for the RMM ports to the customer's network, Ports 51 and/or 52 in Turtle are linked to the customer's network either directly or through the 10Gb Top-of-the-rack switches. The RMM port is used by EMC field service personnel for monitoring, diagnosis and installation.

Customer applications connect to ECS by using the (10Gb) public IP addresses of the ECS nodes. Customers can leverage their own load balancer to provide load balancing across the ECS nodes. Geo-replication (WAN traffic) also uses the customer's data network.

1.1.3 Upgrading ECS

You can upgrade ECS by adding extra disks to existing DAEs or adding pairs of servers and DAEs (4 servers at a time). Each DAE can store maximum 60 disk drives, but disks are added 15 at a time as shown below in the diagrams. Check with EMC's sales tools to verify what upgrades are possible for a given customer configuration.



ECS OS as well as the Fabric can also be upgraded or patched (minor upgrade). This will be done by EMC professionals, and sometimes this process can be disruptive.

As mentioned in Section 2.1.1, an ECS rack can also be expanded by daisy-chaining one or more ECS racks to an existing rack. The 1Gb switches in the racks are used for serially linking the racks. The 10Gb switches in each rack will be connected to a customer-provided switch or backplane. Thus the data and chunk traffic will flow through the 10Gb network.

1.1.4 DIY (Do It Yourself ECS)

ECS Software can be installed in EMC-approved 3rd-party commodity hardware. Today, ECS is certified for HP SL4540, but more options will be available in the future.

HP SL4540 server can have a maximum of 60 3.5" disk drives attached to a single server and this is similar to one ECS node. HP SL4540 has 64GB RAM and one internal 3.5" disk drive (1TB or 4TB).

1.1.5 Requirements for the Installation of ECS

This section describes the pre-requisites to install an ECS. Unlike the older versions of ECS, starting with 2.0, we don't need external virtual machines to manage or install ECS. With the new "installer-less" and "controller-less" ECS, installation is much easier and is all done through the ECS nodes themselves.

First, the customer has to prepare their infrastructure to have available servers to provide services such as DHCP, DNS, NTP etc. There must also be pre-planning to allocate and map specific IP addresses and hostnames to both sets of ECS nodes and RMM ports. Both forward and reverse nslookup mappings must be done as well.

The ECS switches must be properly connected to the customer's network so that the ECS nodes have access to the network services. In practice, there can be a little bit of effort to make ECS switches talk to the customer's network, but it's not difficult. It's a good idea to provide the configuration files of ECS switches to the customer's network admin. It is good to remember that the two 10Gb switches are linked via MLAG and thus appear as one big switch. The easiest way to test the connections is to boot the first ECS node, get the MAC address of the 10Gb port, and request the customer to assign the right IP address to that MAC address in their DHCP server. Now if the node gets the right IP address (may require a reboot), then the network connectivity is good.

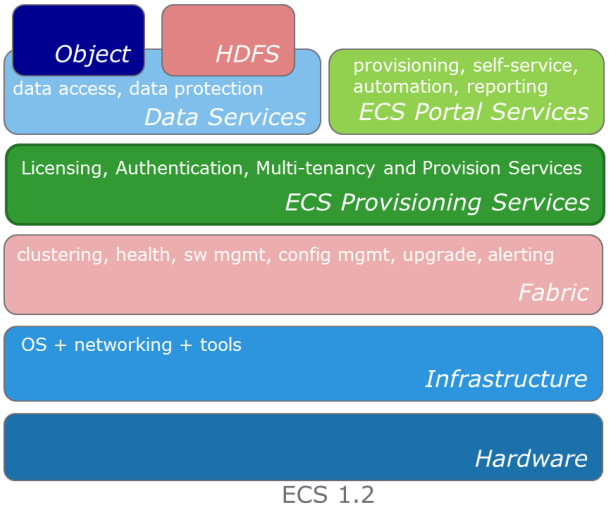
However, if the network issues persist, there is a way to use static IP addresses to proceed further -- one would use *ipmitool* command to configure the RMM ports and setrackinfo command to configure the 10Gb (public) data ports.

1.2 ECS Software Architecture

ECS is made up of four layers, as shown on the right. The Linux Operating System runs on the EMC hardware or a certified 3rd party infrastructure. Tools for networking and commands for ECS are also implemented in this layer.

The Fabric layer provides clustering, health, software management, configuration management, upgrade capabilities and alerting.

In ECS 2.0 and beyond, the controller management component – known as the "Element manager" – runs inside the ECS node. The function of the Element Manager is equivalent to ViPR Controller, but the user interface is slightly different. In the diagram to the right, it is shown as "ECS Provisioning Services."



ECS 1.2

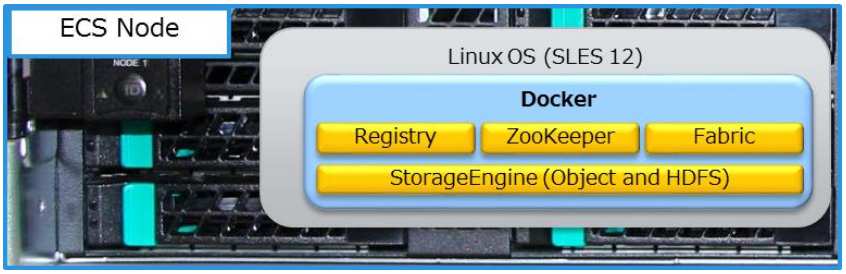
An application can access the data as object and/or HDFS through the Data Services layer. Access by NFS will be available in the future. Data access and data protection are provided by the patented storage engine of ECS.

1.2.1 Infrastructure (OS)

ECS Appliance uses SUSE Linux Enterprise Server 12 (Kernel version 3.12) as the base ECS operating system.

1.2.2 Fabric

ECS uses Docker containers to provide various Fabric services, as shown in the picture below. The operating system and applications are isolated by using Docker. Consequently, development, test and production environments can be run in the same manner without environment dependency. This also simplifies installation and upgrade of the application. The ECS Fabric service itself is run as a Docker container. The ECs Fabric service runs on all nodes in the ECS Appliance and contains the connectivity information. The Registry contains all ECS images and is used during installation, upgrades, and node replacement. The Zookeeper maintains the cluster configuration information.



1.2.3 Storage Engine

ECS Data Services are provided by the storage engine which is packaged in a Docker container and installed on each node, thus providing a distributed and shared service (see picture below). This architecture also enables global namespace, management across geographically dispersed data centers, and geo-replication. The storage engine is composed of several independent layers that represent major features and functionalities. Each layer is independently scalable, highly available, and has no single point of failure.

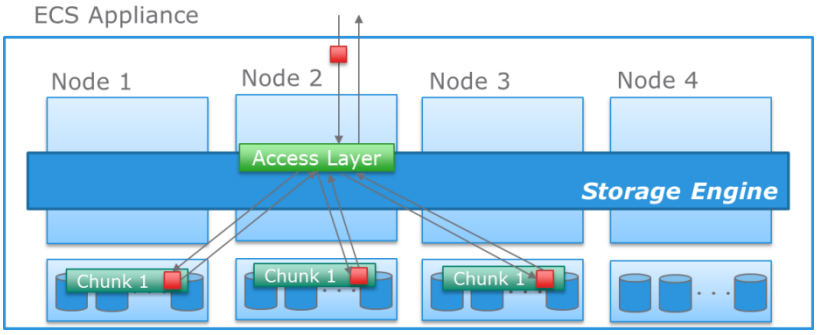
Transaction Layer	• Parses object request. • Reads / writes object data to chunk.
Index Layer	• File-name/data-range to chunk mapping • Secondary indices.
Chunk Management Layer	• Chunk information (e.g. location) • Per chunk operations.
Storage Server Management Layer	• Monitors the storage server & disks. • Re-protection on hardware failures.
Partitions Record Layer	• Records owner node of a partition. • Records btree and journal of partitions.
Storage Server (Chunk I/O) Layer	• Direct I/O operations to the disks.

1.2.4 Chunk

All types of data including object, meta-data and index are stored in "chunks". A chunk is a 128MB logical container of contiguous space. Note that each chunk can have data from different objects and ECS uses indexing to keep track of all the parts of an object which may be spread across different chunks. Chunks are written in an append-only pattern -- meaning, an application cannot modify/delete existing data within a chunk. Therefore, there is no locking required for I/O, and no cache invalidation is required either, offering superior performance. Further, by virtue of being an append-only system, ECS has built-in journaling, snapshot and versioning.

A chunk can be accessed by any of the ECS nodes. In the diagram below, note that the chunks are initially written to multiple nodes which provide data protection in case a drive, node or multiple nodes fail. Then later on, Erasure Coding is performed on the data and spread across multiple nodes and disks.

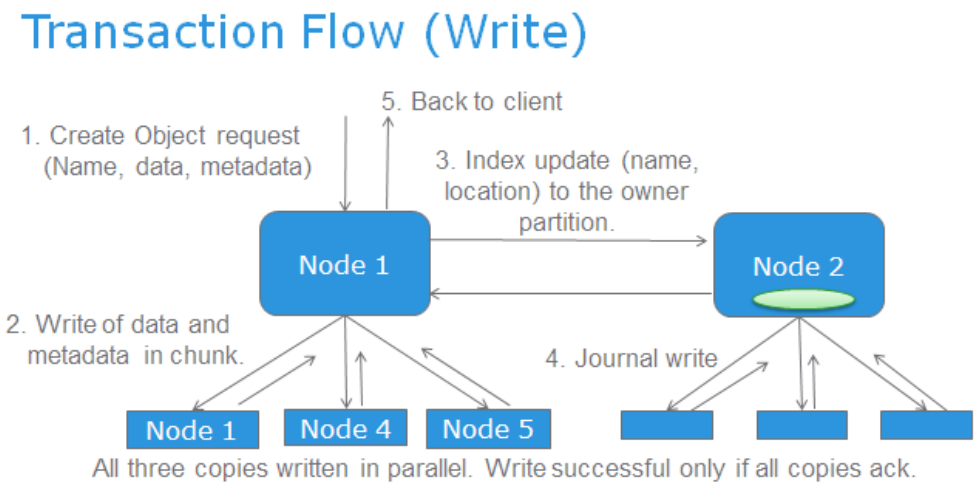
Note that ECS also compresses all the data - if the data is compressible - that are written to the disk.



1.2.5 Read and Write Transactions

This transaction flow illustrates a protected write.

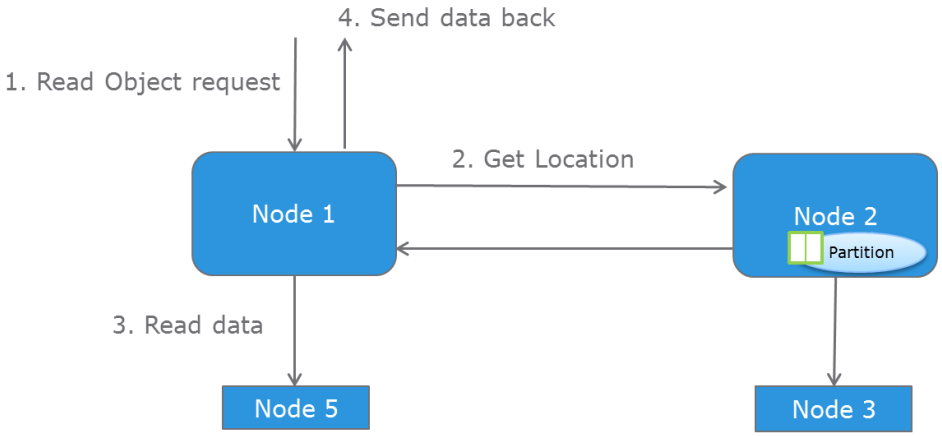
1. The storage engine receives a Create Object request
2. ECS writes the data and metadata to three chunks in parallel (if the compressed object < 128MB)
3. Once the 3 copies are acknowledged, ECS writes to the index with name and location
4. The partition owner ("Node 2" in this case) writes the Journal to 3 chunks
5. Acknowledgement is sent to the client



On the other hand, a Read request is simple (see diagram below):

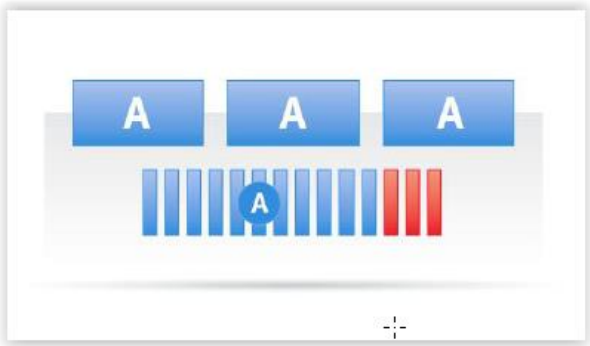
1. The system received a Read object request
2. ECS gets the location from the index in the partition owner
3. Reads the data

4. Sends data to the client



1.2.6 Erasure Coding

ECS employs erasure coding for enhanced data protection with lower overhead as compared to commonly deployed conventional data protection mechanisms, e.g., RAID. The ECS storage engine implements the Reed Solomon 12/4 erasure-coding scheme in which a chunk is broken into 12 data fragments and 4 coding fragments. The resulting 16 fragments are dispersed across nodes at the local site. The data and coding fragments of each chunk are equally distributed across the nodes of the cluster. Thus if there are 8 nodes, each node will have 2 fragments (out of total 16). The storage engine can reconstruct a chunk from any 12 of the 16 fragments.



All data in ECS is erasure coded except the index. The index provides location to objects and chunks and requires to be frequently accessed hence it is always kept in triple mirrored chunks for protection and also cached in memory.

ECS requires a minimum of four nodes to be running the object service at a single site. Erasure coding is implemented at the chunk level and as a background process and the data is triple mirrored at ingest. After erasure coding is complete, the mirrored copies are discarded and a single erasure coded copy persists. In the erasure coded copy, the original chunk data remains as a single copy that consists of 16 data fragments dispersed throughout the cluster, i.e., ECS can read that chunk directly without any decoding or reconstruction. For small objects, ECS doesn't need to read all the fragments of the data -- it can directly read the object from the fragment which contains it. ECS only uses the code fragments for chunk reconstruction when a failure occurs. With ECS 2.0, for objects greater than 128MB in size, erasure coding is done in-line and erasure-coded data is directly written to the disk. Obviously, this is done to enhance performance and decrease disk usage.

In summary, erasure coding provides enhanced data protection from local failures -- for example, disk or node failure -- in a storage efficient fashion as compared to conventional protection mechanisms.

1.2.7 Garbage Collection

ECS is an append-only system which always results in unused blocks of data. When a container is empty, the unused chunk is reclaimed by a background task, sometimes referred to as garbage collection process. If the system reaches a threshold for high capacity (70%), then new chunks are not allocated on fresh contiguous space. Instead, the "garbage holes" – data that has been modified and thus has become invalid – in existing chunk are used. Erasure coding is re-calculated using the new data. The holes are now ready to be overwritten. This is an efficient way to achieve space reutilization without resorting to bulk movement of data.

1.2.8 ECS Portal (GUI) Components

While logged into ECS Portal, the administrator will configure and use several components, and this section describes those technical terms/components.

Virtual Data Center (VDC)

A VDC is essentially a complete ECS system in a single or multiple racks. Within the Portal, the administrator defines a VDC by using the IP addresses of all the nodes in the system. Thus an ECS with 8 nodes in a single rack will be defined as a VDC using the IP addresses of all the 8 nodes.

When racks are combined and placed into a VDC together, they must be daisy chained to one another through the management switch. The daisy chained network of multiple ECS Appliances is referred to as the Nile Area Network (NAN). When multiple racks are combined together within a VDC through the NAN, management and capacity of all the nodes in all the racks are combined and managed as if they were in a single enormous rack (see Section 3.2 for details and a picture for how multi-rack systems are configured).

In the Portal, go to Manage -> Virtual Data Centers to view VDC details, to create a new VDC, to modify existing VDCs, to delete VDCs and to federate multiple VDCs for a multi-site deployment.

Storage Pool

A Storage Pool can be thought of as a subset of nodes belonging to a VDC. An ECS node can belong to only one storage pool; a storage pool can have any number of nodes, the minimum being 4. A storage pool can be used as a tool for physically separating data belonging to different applications. The first storage pool that is created is known as the "system storage pool" because it stores system metadata. The system storage pool cannot be deleted. In the Portal, go to Manage -> Storage Pools to view the details of existing storage pools, to create new storage pools, to modify existing storage pools, and to delete storage pools.

Replication Group

A replication group is a logical collection of storage pools from a single site or multiple sites. In other words, replication groups can be local (single site) or global (multiple sites). Local replication groups protect objects within the same VDC against disk or node failures. Global replication groups protect objects against disk, node, and site failures. The choice of storage pools also determines how data will be replicated across sites. Rather than having to choose a "source" and a "target" for replications, you choose storage pools from various sites, and this will automatically enable replication among those sites. For example, if you create a replication group with three storage pools from three different sites, objects from the first site will either be replicated to the second or the third site. Similarly, an object written to the second site will be replicated either to the first or the third site.

Namespace

A namespace -- same concept as a "tenant" -- is a logical construct that is mapped to a particular replication group. The key characteristic of a namespace is that users from one namespace cannot access objects belonging to another namespace. Multiple namespaces can be mapped to a single replication group.

Namespaces are global resources in ECS and a System Admin or Namespace Admin accessing ECS at any linked VDC can configure the namespace settings. In addition, object users assigned to a namespace are global and can access the object store from any linked VDC.

An ECS namespace has the following attributes that are configured within the Portal:

Default Replication Group: The replication group in which a bucket will be created if no replication group is specified in a request.

Namespace Administrators: These are users assigned to the "Namespace Admin" role for the namespace. The Namespace Admin is an ECS management user and can be a local or domain user.

User Mappings: The domains, groups, and attributes that identify the users that can access the namespace. They are also called "Namespace Users" and they can log into the ECS to administer their own buckets (described in the "bucket" section below).

Allowed (and Disallowed) Replication Groups: The REST API enables a client to specify which replication groups can be used by the namespace. It is also possible to specify retention policies and specify a quota for the namespace.

Bucket

Buckets are created on top of namespaces to give applications access to ECS. To understand the concept of buckets, remember that containers are required to store object data. In S3, these containers are called "buckets" and this term has been adopted as a general term in ECS. In Atmos, the equivalent of a bucket is a "subtenant"; in Swift, the equivalent of a bucket is a "container", and for CAS, a bucket is a "CAS pool". Buckets are global resources in ECS and belong to a replication group. Where the replication group spans multiple sites, the bucket is similarly replicated across the sites.

The object application will connect to the IP address of one of the nodes within the namespace containing the desired bucket. Depending on whether the application will use the S3 protocol, Swift protocol, or Atmos protocol, the application will use a different port on the IP address to which it is going to connect. Once the application has connected, it will read or write the selected object. When writing an object, the rack to which the application is connected will "own" the object. After the object is written, applications can read the object by connecting to any of the racks within the namespace containing the bucket, but the process by which the object is returned to the application varies depending on which rack the application is connected to when it retrieves the object. This process is described in details later in this document in Section 4 (Geo-Federation).

Users

ECS requires two types of user: management users who can perform administration of ECS, and object users who access the object store to read and write objects and buckets using the supported data access protocols (S3, EMC Atmos, OpenStack Swift, and CAS). Management users can access the ECS portal. Object users cannot access the ECS portal but can access the object store using clients that support the ECS data access protocols. Management users and object users are stored in different tables and their credentials are different. Management users require a local username and password, or a link to a domain user account. Object users require a username and a secret key. Hence you can create a management user and an object user with the same name, but they are effectively different users as their credentials are different.

1.2.9 Perform test I/O to an ECS

The most common way to read from and write to an ECS is to use an S3-compatible application. With a newly installed ECS, a quick way to do simple I/O is to use a freeware such as the S3 Browser.

To summarize the steps: Go through the ECS Portal to create a Namespace, and then use the S3 Browser to access ECS.

Step 1: Provision a Namespace via ECS Portal

There are a few easy steps to accomplish this:

- First, create a Storage Pool in the ECS Portal by choosing the ECS nodes and giving that collection a name.

- Next, from the Virtual Data Center (VDC) tab on the left
    - Click on "Get VDC Access Key"
    - Copy the displayed key
    - Click on "New Virtual Data Center" and create a VDC using the above key and the IP addresses of all the ECS nodes.
- Then create Replication Group using the newly created Storage Pool and the VDC.
- Now, create a Namespace using "root" and the Replication Group (created in Step 3). Namespace is analogous to a "Tenant." This is where buckets will be created.
- At this point, you have an option of moving to the next step or creating a bucket. If you create a bucket, you will see that bucket after configuring the S3 browser. To create a bucket, go to the "Buckets" tab, choose the Namespace that you just created from a dropdown, and give the bucket a name.
- Finally, go to Users, click on "Object Users" □ "New Object users"; choose the Namespace for "root" and click on "Next to add Passwords". In that next page, generate an S3 secret key for root and copy it to a text editor (to be later used within S3 Browser).

You're done with steps required in ECS Portal.

Step 2: Configure S3 Browser and Access ECS

Now, open the S3 Browser. Go to "Accounts" on the top left and choose "Add New Account". There are four things that need to be filled out or chosen:

- Type in an Account name
- Under "Storage Type", Choose S3 Compatible Storage
- In "REST Endpoint", type in the IP Address of one of the ECS nodes along with the port of 9020 (9021 is for secure https). The format is <IP_address>:<port_number>
- Under "Access Key Id", put in "root"
- Copy and paste the secret key from Step 1-E into the field for "Secret Access Key".
- Save Changes and you should be good to go.
- Now you can click on "New bucket" in the main page and upload files to that bucket from your local drive. If you had created a bucket from within the ECS Portal at the end of Step 1-D, you will now see the bucket listed here as well.

2 ECS INFRASTRUCTURE BEST PRACTICES

2.1.1 ECS Network Setup

As mentioned in Section 2.1.2, the ECS rack will have three switches – two 10Gb Arista switches and one 1Gb Arista switch. The 10Gb switches, given the nicknames of Hare and Turtle, are used to transfer a high amount of data between customer applications and the ECS nodes.

Within the ECS rack, cables of different colors are used for different purposes to make cabling easy and to avoid any mistakes. Black, blue, gray and white cables are used for various network connections.

2.1.2 Hare and Rabbit – 10Gb Switches

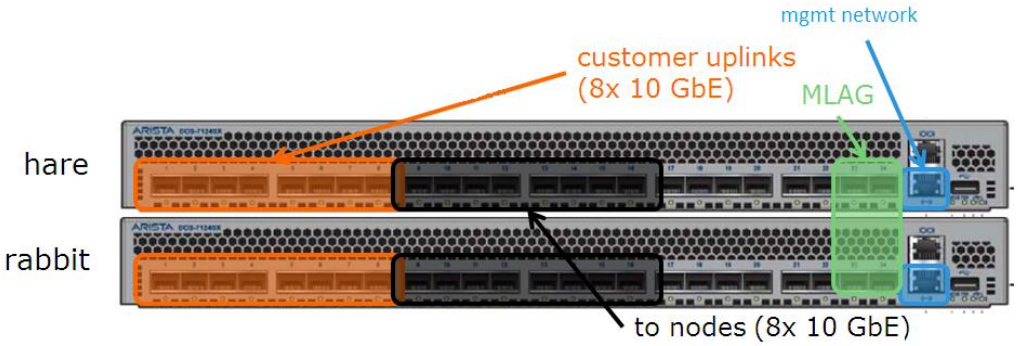
These are 24-port Arista switches. As shown in Diagram 1 below, ports 1 thru 8 of each 10Gb switch are the "uplinks" ports that connect to the customer network for bi-directional data transfer between customer applications and the switch. Depending on bandwidth requirement, the customer can choose anywhere from 1 to 8 ports on each of the switches.

Ports 9-16 of each switch are connected to the ECS nodes using black cables. In a typical full ECS rack, there are 8 ECS nodes, each one with two 10Gb connections. Thus each ECS node will connect to both of these 10Gb switches in the designated ports.

Each last port on the far right of Hare and Rabbit is connected to the 1Gb switch (Turtle). This is shown in the diagram below as being part of the management network. Thus, to make any configuration changes to the 10Gb switches, one would go through the 1Gb switch.

Ports 23 and 24 in these switches are part of the Multi-Link Aggregation (MLAG). Port 23 of Hare is connected to Port 23 of Rabbit; port 24 of Hare is connected to port 24 of Rabbit. The switches are said to be an "MLAG pair." MLAG is a feature that logically links the switches and enables active-active paths between the nodes and customer application. This results in higher bandwidth while preserving resiliency and redundancy in data path. Any networking device supporting static LAG or IEEE 802.3ad LACP can connect to this MLAG pair. Finally, because of MLAG, these two switches appear and act as one large switch.

DIAGRAM 1: Showing the ports and their uses in the 10Gb switches



2.1.3 1Gb Switch – Turtle

The management network uses a 48-port 1GbE Arista switch – internally known as Turtle. This switch provides access to management of ECS nodes and the other switches in the rack; this is also the switch for internal traffic between the ECS nodes. Port connections are shown in the two diagrams below. Note the use of different colors for cables in the diagram; these also represent the actual colors of the cables used in ECS racks.

Ports 1-24 are dedicated to the private management of the ECS nodes, with one connection per ECS node. In a single rack, the maximum number of ECS nodes is 8. So the reason there are 24 ports dedicated to private management is that ECS nodes from a second rack can be linked to this switch – a configuration that is rarely implemented in the field as of now. These ports are used for traffic that is private to ECS nodes – management communication, data, PXE booting of nodes etc.

The ports that are marked for "RMM Connectivity" – 25 through 48 in Diagram 2 below – are used to manage the ECS nodes using the nodes' RMM addresses. In a typical rack, ports 25-32 are connected to the 8 RMM ports of 8 ECS nodes using gray cables. The RMM port on an ECS node is the far right NIC, as shown in Diagram 3. The RMM ports have their own IP addresses and MAC addresses, and are used during installation and diagnostic procedures.

Ports 49 and 50 of Turtle are each connected to one port in Hare and Rabbit respectively using white cables.

Port 51 connects to the customer's network management switch for RMM access. Port 52 can be daisy-connected to another ECS rack in order to scale an ECS appliance and add more capacity.

DIAGRAM 2: Illustrates the ports and their uses in the 1Gb management switch

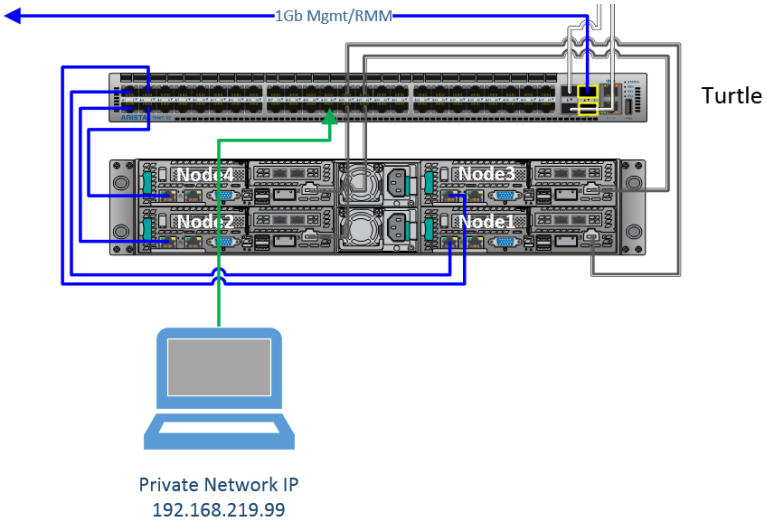
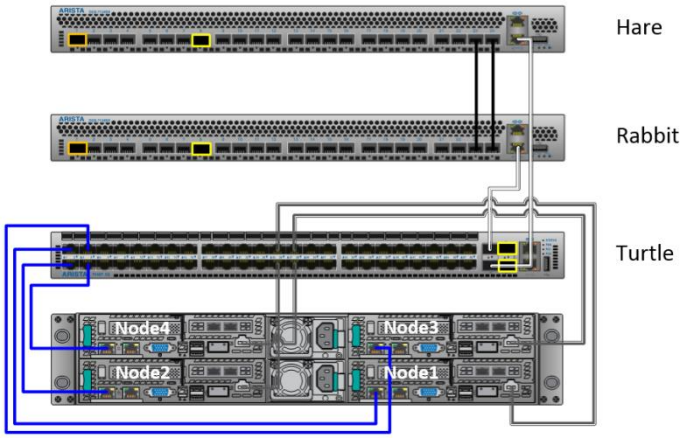


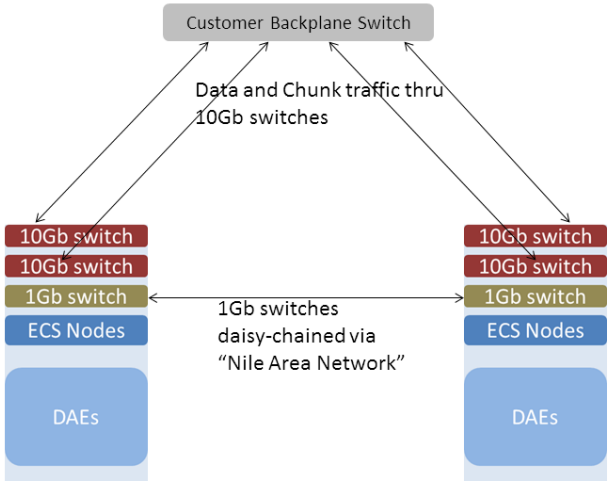
DIAGRAM 3: Illustrates connections between the Turtle (1Gb Switch) and the other components (two 10Gb switches and four



servers). Note: this diagram shows only 4 ECS nodes. If there are 8 ECS nodes in a rack, then four more ports will be taken up for "private management" and four more ports will be used for "RMM Connectivity."

2.2 Multi-Rack Configurations

In a multi-rack configuration, all cross-rack aggregation of the 10Gb switches is done via customer-designed and customer-provided switches as shown in the picture below. One to eight uplinks per 10Gb switch (16/rack) are used. As for the management network, one can "daisy-chain" the management networks or route the switches via customer network. However,



additional 1Gb switches will be needed.

The link through the 1Gb switches enables the ECS software to treat the multiple racks as a single ECS system with shared resources. Note that here are a couple of different topology options to inter-connect the 1Gb management switches.

Extra racks can be added to an existing ECS. Any new data that gets written to the ECS will include the new additional nodes, DAEs and disks to spread out the data and chunks for increased availability and redundancy.

2.3 Load Balancers

A load balancer is a critical piece to a ViPR Data Services / ECS installation in order to evenly distribute load across all of the data services nodes. Load balancers can use simple algorithms such as random choice or round robin. More sophisticated load balancers may take additional factors into account, such as a server's reported load, response times, up/down status, number of active connections, geographic location and so on. The customer is responsible for implementing load balancers; customers have several options including Manual IP allocation, DNS Round Robin, Client-Side Load Balancing, Load Balancer Appliances, and Geographic Load Balancers. The following are brief descriptions of each of those methods.

Manual IP Allocation

Manual IP allocation means that the data node IP addresses are manually distributed to applications. This is not recommended because it does not evenly distribute load between the nodes and does not provide any fault-tolerance if a node fails.

DNS Round-Robin

With DNS Round-Robin, a DNS name is created for ECS and includes all of the IP addresses for the data nodes. The DNS server will randomly return the IP addresses when queried and provides some pseudo-load balancing. This generally does not provide fault-tolerance because you would need to remove the IP addresses from DNS to keep them out of rotation. Even after removing them, there is generally some TTL (time-to-live) issues where there will be a delay to propagate the removal. Also, some operating systems like Windows will cache DNS lookups and can cause "stickiness" where a client will keep binding to the same IP address, reducing the amount of load distribution to the data nodes.

Software (client-side) load balancing via ECS Smart Clients

The ECS Java SDK (and eventually more) supports client-side load balancing by using the S3 "endpoints" method to query the IP addresses of the data nodes and then implementing an in-process load balancer to distribute requests across the nodes. It also tracks node state to remove nodes from the rotation that are having connectivity issues or producing 500 errors.

The downsides to the smart client approach are:

- Only works if all of the data nodes are directly IP-routable.
- Does not work with NAT, and only works with the S3 protocol

Hardware load balancing

Hardware load balancing is the most common approach to load balancing. In this mode, an appliance (hardware or software) receives the HTTP request and forwards it on to the data nodes. The appliance keeps track of the state of all of the data nodes (up/down, # of connections) and can intelligently distribute load amongst the nodes. Generally, the appliance will proactively "health check" the node (e.g. GET/?ping on the S3 head) to ensure the node is up and available. If the node becomes unavailable it will immediately be removed from rotation until it passes a health check.

Another advantage to hardware load balancing is SSL termination. You can install the SSL certificate on the load balancer and have the load balancer handle the SSL negotiation. The connection between the load balancer and the data node is then unencrypted. This reduces the load on the data nodes because they do not have to handle the CPU-intensive task of SSL negotiation.

The downside to hardware load balancing is generally cost: you have to provision hardware and possibly purchase software licenses, support, etc.

Geographic load balancing

Geographic load balancing takes Hardware Load Balancing one step further: it adds load balancing into the DNS infrastructure. When the DNS lookups occur, they are routed via an "NS" record in DNS to delegate the lookups to a load balancing appliance like the Riverbed SteelApp. The load balancer can then use Geo-IP or some other mechanism to determine which site to route the client to. If a site is detected to be down, the site can be removed quickly from DNS and traffic will be routed to surviving sites.

The downside again is cost: you generally have to purchase advanced licenses to get geographic load balancing on top of whatever appliance you're using.

3 GEO-FEDERATION AND GEO-REPLICATION

3.1 Geo-Federation

In essence, geo-federation means managing a geographically distributed environment as a single logical resource. Inside ECS, the term refers to the ability to federate multiple sites or Virtual Data Centers (VDCs). The obvious benefits are ease of management and the ability to use resources from multiple data centers.

To manage or monitor a single site, the administrator simply logs into one of the nodes in the VDC. This is equivalent to having a single-site federation. The administrator can subsequently add other sites/VDCs to the federation by navigating to Manage→VDC in the ECS portal at the initial site. Further, customer can choose what data is replicated across sites for enhanced data protection.

To perform some administrative tasks such as creating storage pools or viewing performance statistics, one has to log into each site/VDC individually.

3.2 Geo-Replication

As discussed in Section 1.2.6, ECS offers erasure coding to help provide enhanced data durability without the overhead of storing multiple copies of the data. However, this does not protect against site failures/outages. Geo-replication provides enhanced protection against site failures by having multiple copies of the data, i.e., a primary copy of the data at the original site and a secondary copy of the data at remote site/VDC. Both the primary and the secondary copy of the data are individually protected via erasure coding. This means each copy has protection from local failures, e.g., disk or node failure.

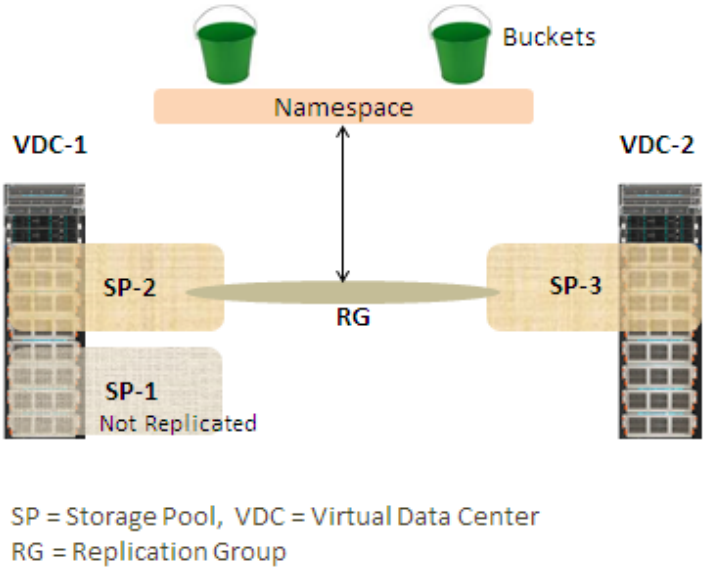
Geo-replication ensures that the data is protected against site failures/disasters. Also, unlike other solution in the market place, ECS does not generate WAN traffic while recovering from local failures, e.g., disk failure. ECS gives customers the option to link geographically dispersed systems and bi-directionally replicate data among these sites across WAN. As will be discussed below, several smart strategies such as geo-caching as well as accessing the physically closest array are used to reduce WAN traffic for data access. ECS also uses compression to reduce the WAN bandwidth and cost.

3.2.1 Geo-Replication Architecture

Replication Group (RG) is the logical container that defines what namespace maps to what physical resources i.e. Storage Pool(s), to store the data for the namespace. A single-site RG defines the Storage Pool where the data mapped to that RG resides. A multi-site RG defines a set of storage pools amongst where the primary and the secondary copy of the data resides. Further, the primary copy of the data resides in the site the write came into and the other storage pools jointly store a single secondary copy of the data.

The picture below gives a quick overview of the mapping from the logical model to the physical infrastructure, i.e., Storage Pool. A Customer's applications access a bucket which belongs to a Namespace. One or more Namespaces are mapped to a Replication

Group which is a logical grouping up of Storage Pools from different Virtual data Centers. This allows data service applications to read/write objects without having to worry about where (in which data center or storage array) the data resides.



To summarize the steps involved in creating geo-replication among the ECS systems shown in the picture above:

1. Log into an ECS system (in this case, one of the nodes of VDC-1)
2. Create a Storage Pool (SP-2) by using a certain number of physical ECS nodes (minimum 4)
3. Create a local VDC (VDC-1) using the VDC Key and the IP addresses of all the ECS nodes.
4. Log into the other site and repeat steps 2, i.e., create a Storage Pool (SP-3).
5. Log back into VDC-1 and navigate to Manage→Virtual Data Center and add the remote VDC using the VDC Key and the IP addresses of all the ECS nodes in VDC-2. This creates a 2-site geo federation.
6. Create a Replication Group that comprises of VDC1:SP2 and VDC2:SP3.
7. Create a Namespace and map the RG created in Step 6.
8. Create buckets using the above Namespace

Note that the ECS Portal itself does not have a way to read or write to the buckets. The customer's applications have to use S3, Atmos, CAS API or HDFS to access the buckets. For quick demo or testing purposes, one can use freeware such as the "S3 Browser" to create buckets or read from and write to the buckets.

3.2.2 Geo-Replication XOR

In a geo-federated setup with multiple sites/ VDCs with geo-replication configured, ECS will replicate chunks from the primary VDC to a remote site in order to provide high availability. However, in a multi-site environment, this simple replication can lead to a large overhead of disk space. To prevent this, ECS uses a smart technique to reduce the overhead while preserving the high availability features. This can be illustrated with a simple example as detailed below.

Consider 3 VDC's in a multi-site environment - VDC1, VDC2 and VDC3, and that VDC1 has chunk C1 and VDC2 has chunk C2. With simple replication, a secondary copy of C1 and a secondary copy of C2 may be placed in VDC3. Since all chunks are of the same size, this will result in a total of 4 x 128MB of space being used to store 2 x 128MB of objects.

In this situation ECS can perform an XOR operation of C1 and C2 (mathematically, written as C1 $\oplus$ C2) and place it in VDC3 and get rid of individual secondary copies of C1 and C2. Thus, rather than using 2 x 128MB of space in VDC3, ECS now uses only 128MB (the XOR operation results in a new chunk of the same size).

In this case, if VDC1 goes down, we can reconstruct C1 by using C2 from VDC2 and the (C1 $\oplus$ C2) data from VDC3. Similarly, if VDC2 goes down, we can reconstruct C2 by using C1 from VDC1 and the (C1 $\oplus$ C2) data from VDC3.

Counterintuitively, as the number of linked sites increase, The ECS algorithm is more efficient in reducing the overhead. The following table demonstrates this:

Number of sites	Storage Overhead
1	1.33
2	2.67
3	2.00
4	1.77
5	1.67
6	1.60
7	1.55
8	1.52

3.2.3 Geo-Caching

Customers with multi-site access patterns could experience slow performance if data is always fetched from primary site, i.e., where the data was originally written from. Consider a geo-federated environment with Sites 1, 2 and 3. Further, there is an object Object A that was written to from Site 1 and the secondary copy of the object resides at Site 2. At this point a read for the object at Site 3 needs to fetch the object data from either Site 1 or Site 2 to honor the read. This leads to elongated response time, especially in environments with multi-site access patterns. ECS alleviates the response time impact by using some pre-designated percentage of disk space to cache objects that do not exist locally. For frequently accessed objects you would see an reduced response time after the initial copy of the object has been cached locally. While the data replication is asynchronous, the metadata replication is synchronous and ensures that the system never responds to a read at a remote site with a stale version of the data, thereby being true to the tenant of strong consistency.

The cache is a LRU implementation and cache size is adjusted when nodes/disks are added to the storage pool.

3.2.4 Temporary Site Outage a.k.a Acess During Outage

Temporary site failure refers to either a failure of WAN connection between two sites or a failure of an entire site itself (such as a natural disaster or power failure). ECS has the ability to detect and automatically handle any such temporary site failures. VDCs in a geo-federated environment establish a heartbeat mechanism. Sustained loss of heartbeats for preset duration is indicative of a network outage and the system adjusts its behavior accordingly. Specifically, without the TSO (Temporary Site Outage) functionality, the inability to communicate with the node that owns the object (from a metadata perspective) leads to the system rejecting the reads and the writes to the object. This is in line with the principle of strong consistency that ECS system adheres to.

With Access During Outage enabled on a bucket and upon detecting a temporary outage, the system reverts to an eventual consistency model, i.e., reads/writes from secondary (non-owner) site are accepted and honored. Further, a write to a secondary site during a network outage causes the secondary site to take ownership of the object. This allows each VDC to continue to read and write objects from the buckets in the shared namespace. Finally, the new version of the object will become the authoritative version of the object during reconciliation even if another application updates the object on the "owner" VDC.

Although many object operations continue during a network outage, certain operations are not be permitted, e.g., creation of new buckets, creation of new namespaces, and creation of new users.

When network connectivity between the two VDCs is restored, the heartbeat mechanism will automatically detect this, the service will be restored and objects from the two VDCs will be reconciled. If the same object is updated on both VDC A and VDC B, the copy on the "non-owner" VDC is the authoritative copy. So, if an object that is owned by VDC B is updated on both VDC A and VDC B during synchronization, the copy on VDC A will be the authoritative copy that is kept, and the other copy will be overwritten. The adjoining table captures the list of operations permitted in different system states.

When more than two VDCs are in a replication group and if network connectivity is interrupted between one VDC and the other two, then write/update/ownership operations continue just like they do in configurations of only two VDCS; however, the process for responding to read requests can be a bit more complex and is described below.

If an application requests an object that is owned by a VDC that is not reachable, ECS will send the request to the VDC with the secondary copy of the object. However, the secondary site copy might be XORed and so, secondary site VDC must first retrieve the chunks of the object that were included in the original XOR operation and it must XOR those chunks with the "recovery" copy. This operation will return the contents of the chunk originally stored on the failed VDC. The chunks from the recovered object can then be reassembled and returned. When the chunks are reconstructed, they are also cached so that the VDC can respond more quickly to subsequent requests. Note that this will be time consuming. Thus more VDCs in a replication group imply more chunks that must be retrieved from the other VDCs, and hence more time to reconstruct the object.

Note that if a disaster occurs, an entire VDC can become unrecoverable. ECS treats the unrecoverable VDC as a temporary site failure. If the failure is permanent, system administrator must permanently failover the VDC from the federation to initiate fail over processing which initiates resynchronization and protects the objects stored on the failed VDC. The recovery tasks run as a background process. You can review the recovery process in the ECS Portal: go to Monitor -> Geo Replication -> Failover Processing.

System State Description	Supported Operations (Primary | Secondary)						Consistency
	Read Object; Update Object		Create Object; List Bucket; Edit Bucket		Create Bucket, User, Namespace		
	P	S	P	S	P	S	
Normal All zones connected	✓	✓	✓	✓	✓	✓	Strong
Temp Site Failover Complete System detects temp site outage and completes temp site failover	✓LO	✓AO	✓LB	✓AB	×	×	Concurrent writes transition system to eventual consistency
Site Rejoin Resume normal operation after Incremental Resync. Objects/Buckets updated at both sites are reconciled with the state on secondary as the final version.	✓	✓	✓	✓	✓	✓	Strong

✓LO : Locally owned Objects
✓AO : Acquires Object ownership
✓LB : Locally owned Buckets
✓AB : Acquires Bucket ownership

4 CONCLUSION

Private and hybrid clouds are of great interests to customers as they are facing ever increasing amount of data and the cost associated with data storage in general and public cloud in particular. ECS is a great solution for such customers because of its versatility, hyper scalability, powerful features and use of low-cost commodity hardware. This white paper has covered extensive topics on ECS which include details of hardware, software, networking, architecture, administrative tasks and some best practices. This paper will be useful to EMC engineers as well as customers who are either thinking about purchasing or have just bought an ECS.