Example ILM rules and policies

StorageGRID 11.7

NetApp
January 09, 2024

Table of Contents

Example ILM rules and policies

Example 1: ILM rules and policy for object storage

You can use the following example rules and policy as a starting point when defining an ILM policy to meet your object protection and retention requirements.

> The following ILM rules and policy are only examples. There are many ways to configure ILM rules. Before activating a new policy, simulate the proposed policy to confirm it will work as intended to protect content from loss.

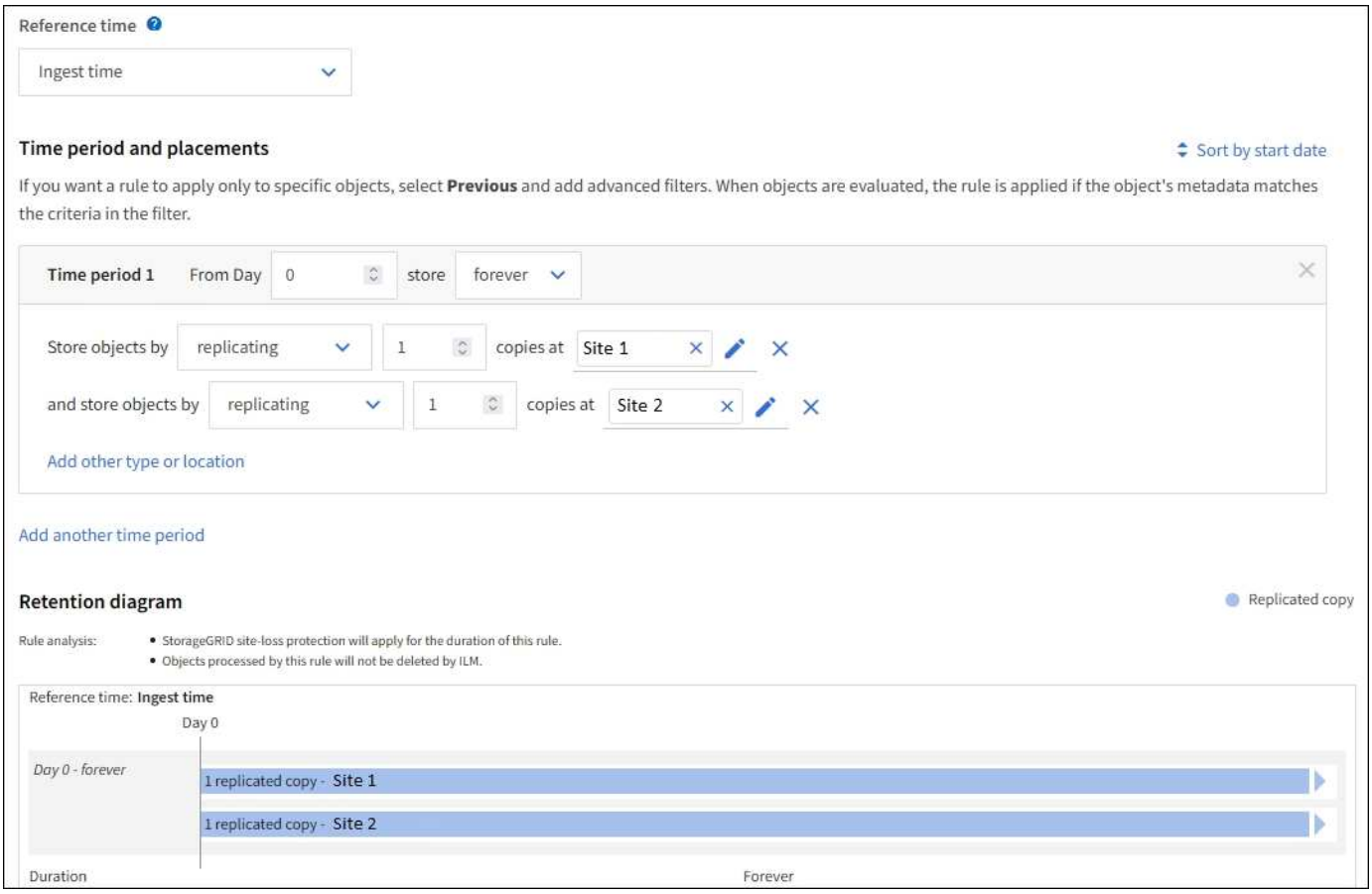
ILM rule 1 for example 1: Copy object data to two sites

This example ILM rule copies object data to storage pools in two sites.

Rule definition	Example value
One-site storage pools	Two storage pools, each containing different sites, named Site 1 and Site 2.
Rule name	Two Copies Two Sites
Reference time	Ingest time
Placements	On Day 0 to forever, keep one replicated copy at Site 1 and one replicated copy at Site 2.

The Rule analysis section of the Retention diagram states:

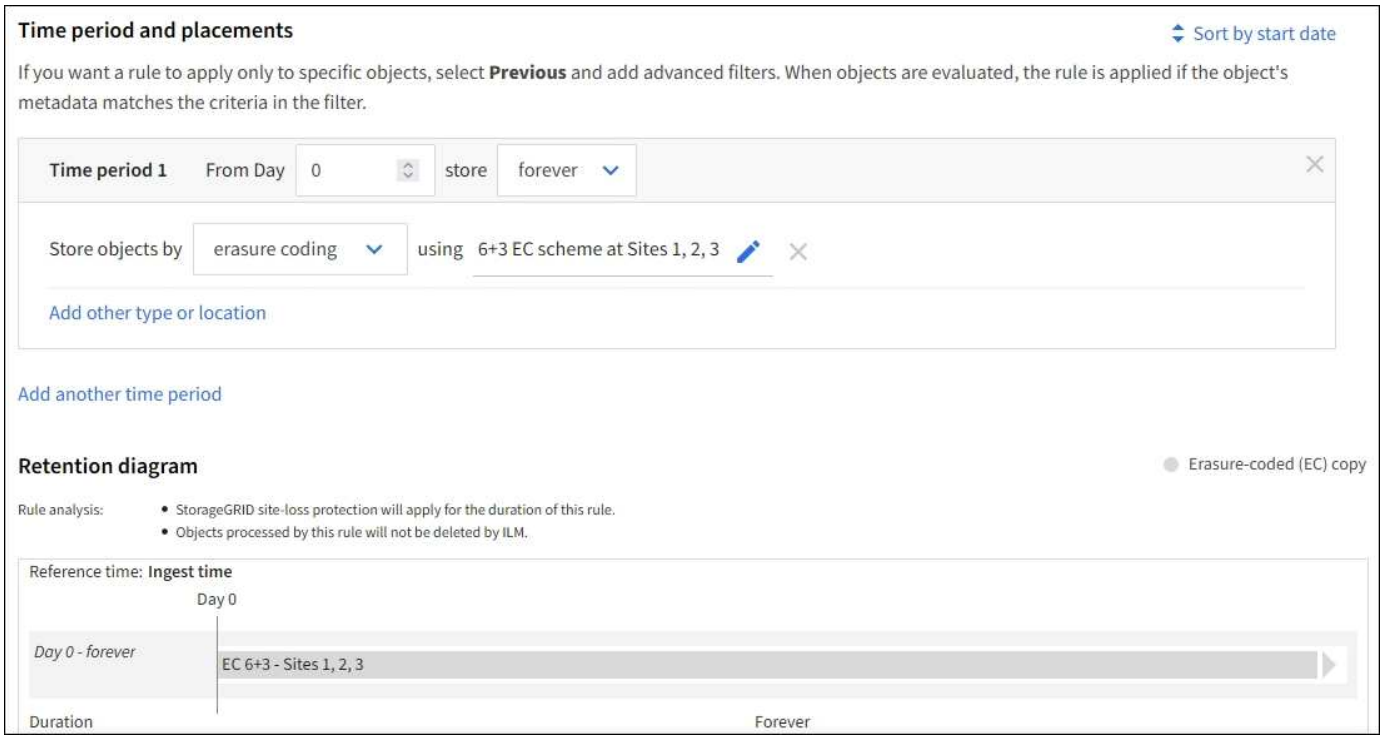
- StorageGRID site-loss protection will apply for the duration of this rule.
- Objects processed by this rule will not be deleted by ILM.



ILM rule 2 for example 1: Erasure coding profile with bucket matching

This example ILM rule uses an erasure coding profile and an S3 bucket to determine where and how long the object is stored.

Rule definition	Example value
Storage pool with multiple sites	• One storage pool across three sites (Sites 1, 2, 3) • Use 6+3 erasure-coding scheme
Rule name	S3 Bucket finance-records
Reference time	Ingest time
Placements	For objects in the S3 bucket named finance-records, create one erasure-coded copy in the pool specified by the erasure coding profile. Keep this copy forever.

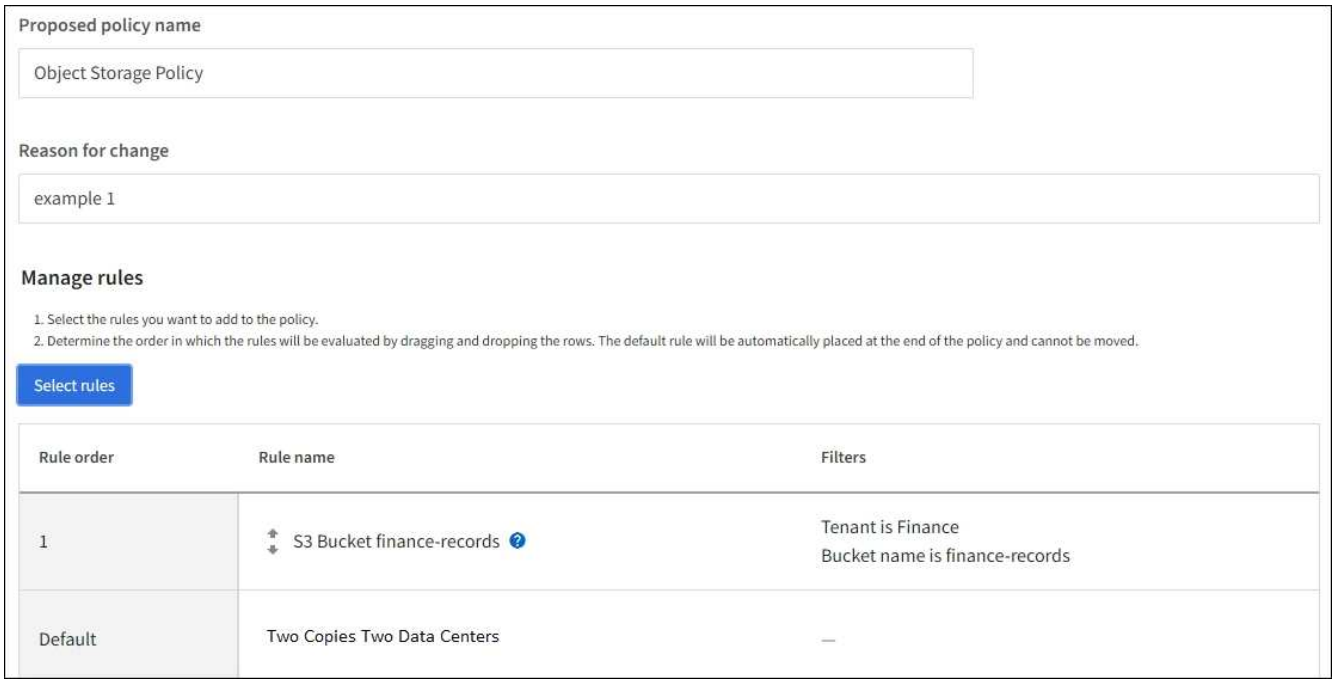


ILM policy for example 1

In practice, most ILM policies are simple, even though the StorageGRID system allows you to design sophisticated and complex ILM policies.

A typical ILM policy for a multi-site grid might include ILM rules such as the following:

- At ingest, store all objects belonging to the S3 bucket named `finance-records` in a storage pool that contains three sites. Use 6+3 erasure coding.
- If an object does not match the first ILM rule, use the policy's default ILM rule, Two Copies Two Data Centers, to store one copy of that object in Site 1, and one copy in Site 2.



Related information

- Create an ILM policy: Overview
- Create a proposed ILM policy

Example 2: ILM rules and policy for EC object size filtering

You can use the following example rules and policy as starting points to define an ILM policy that filters by object size to meet recommended EC requirements.

> The following ILM rules and policy are only examples. There are many ways to configure ILM rules. Before activating a new policy, simulate the proposed policy to confirm it will work as intended to protect content from loss.

ILM rule 1 for example 2: Use EC for objects greater than 1 MB

This example ILM rule erasure codes objects that are greater than 1 MB.

> Erasure coding is best suited for objects greater than 1 MB. Don't use erasure coding for objects smaller than 200 KB to avoid the overhead of managing very small erasure-coded fragments.

Rule definition	Example value
Rule name	EC Only Objects > 1 MB
Reference time	Ingest time
Advanced filter for Object size	Object size greater than 1 MB
Placements	Create a 2+1 erasure-coded copy using three sites

ILM rule 2 for example 2: Two replicated copies

This example ILM rule creates two replicated copies and does not filter by object size. This rule is the default rule for the policy. Because the first rule filters out all objects greater than 1 MB, this rule only applies to objects that are 1 MB or smaller.

Rule definition	Example value
Rule name	Two Replicated Copies
Reference time	Ingest time

Rule definition	Example value
Advanced filter for Object size	None
Placements	On Day 0 to forever, keep one replicated copy at Site 1 and one replicated copy at Site 2.

ILM policy for example 2: Use EC for objects greater than 1 MB

This example ILM policy includes two ILM rules:

- The first rule erasure codes all objects that are greater than 1 MB.
- The second (default) ILM rule creates two replicated copies. Because objects greater than 1 MB have been filtered out by rule 1, rule 2 only applies to objects that are 1 MB or smaller.

Example 3: ILM rules and policy for better protection for image files

You can use the following example rules and policy to ensure that images greater than 1 MB are erasure coded and that two copies are made of smaller images.

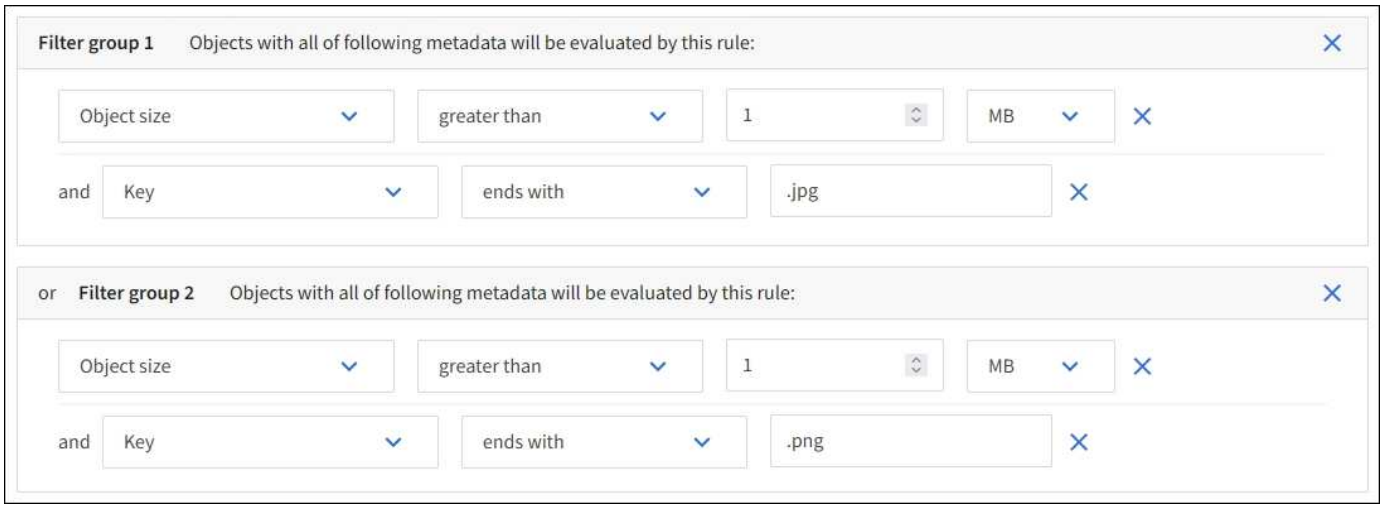
The following ILM rules and policy are only examples. There are many ways to configure ILM rules. Before activating a new policy, simulate the proposed policy to confirm it will work as intended to protect content from loss.

ILM rule 1 for example 3: Use EC for image files greater than 1 MB

This example ILM rule uses advanced filtering to erasure code all image files greater than 1 MB.

Erasure coding is best suited for objects greater than 1 MB. Don’t use erasure coding for objects smaller than 200 KB to avoid the overhead of managing very small erasure-coded fragments.

Rule definition	Example value
Rule name	EC Image Files > 1 MB
Reference time	Ingest time
Advanced filter for Object size	Object size greater than 1 MB
Advanced filters for Key	• Ends with .jpg • Ends with .png
Placements	Create a 2+1 erasure-coded copy using three sites



Because this rule is configured as the first rule in the policy, the erasure-coding placement instruction only applies to .jpg and .png files that are greater than 1 MB.

ILM rule 2 for example 3: Create 2 replicated copies for all remaining image files

This example ILM rule uses advanced filtering to specify that smaller image files be replicated. Because the first rule in the policy has already matched image files greater than 1 MB, this rule applies to image files that are 1 MB or smaller.

Rule definition	Example value
Rule name	2 Copies for Image Files
Reference time	Ingest time
Advanced filters for Key	• Ends with .jpg • Ends with .png
Placements	Create 2 replicated copies in two storage pools

ILM policy for example 3: Better protection for image files

This example ILM policy includes three rules:

- The first rule erasure codes all image files greater than 1 MB.
- The second rule creates two copies of any remaining image files (that is, images that are 1 MB or smaller).
- The default rule applies to all remaining objects (that is, any non-image files).

Rule order	Rule name	Filters
1	EC image files > 1 MB	Object size is greater than 1 MB
2	2 copies for small images	Object size is less than or equal to 200 KB
Default	Default rule	—

Example 4: ILM rules and policy for S3 versioned objects

If you have an S3 bucket with versioning enabled, you can manage the noncurrent object versions by including rules in your ILM policy that use “Noncurrent time” as the reference time.

If you specify a limited retention time for objects, those objects will be deleted permanently after the time period is reached. Make sure you understand how long the objects will be retained.

As this example shows, you can control the amount of storage used by versioned objects by using different placement instructions for noncurrent object versions.

The following ILM rules and policy are only examples. There are many ways to configure ILM rules. Before activating a new policy, simulate the proposed policy to confirm it will work as intended to protect content from loss.

To perform ILM policy simulation on a noncurrent version of an object, you must know the object version’s UUID or CBID. To find the UUID and CBID, use object metadata lookup while the object is still current.

Related information

- How objects are deleted

ILM rule 1 for example 4: Save three copies for 10 years

This example ILM rule stores a copy of each object at three sites for 10 years.

This rule applies to all objects, whether or not they are versioned.

Rule definition	Example value
Storage pools	Three storage pools, each consisting of different data centers, named Site 1, Site 2, and Site 3.
Rule name	Three Copies Ten Years

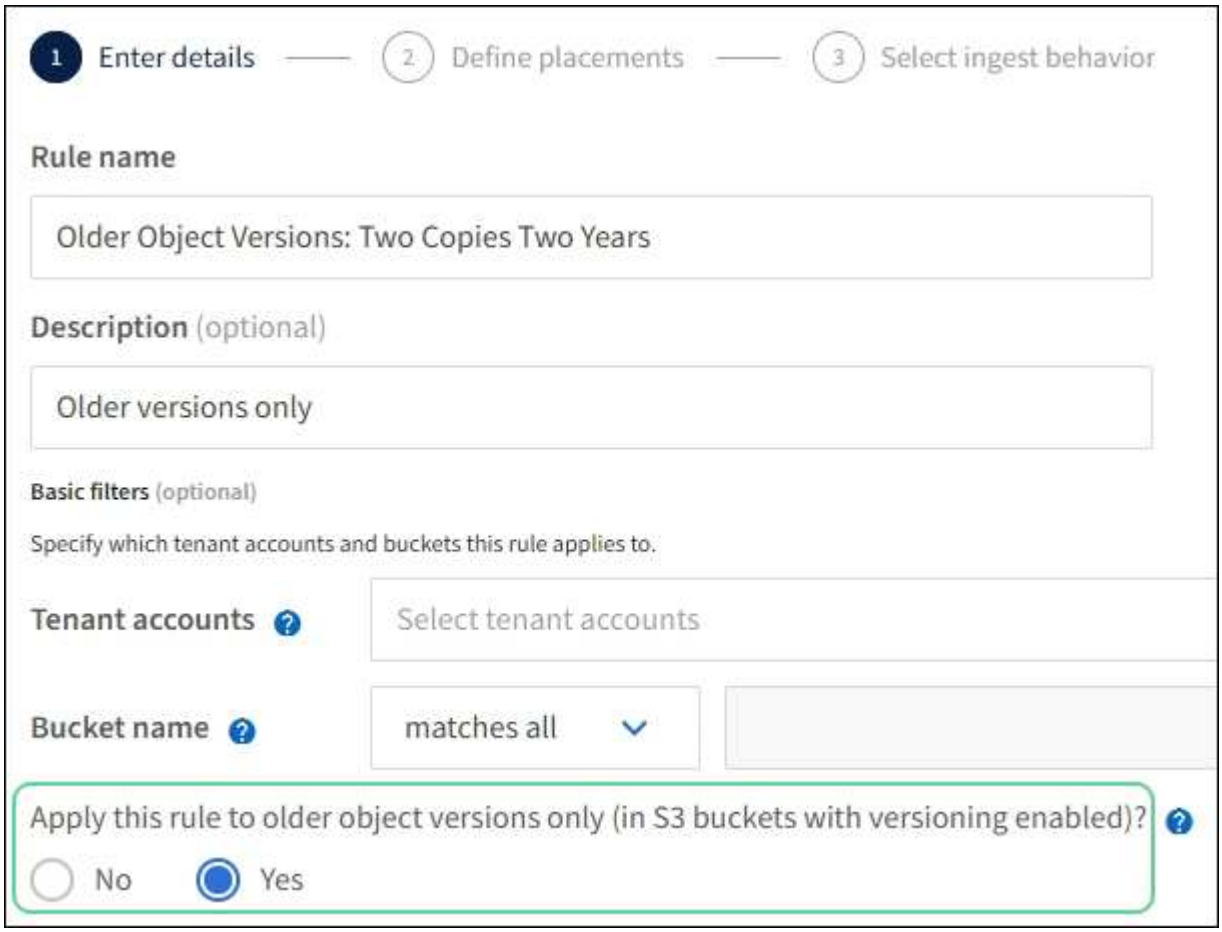
Rule definition	Example value
Reference time	Ingest time
Placements	On Day 0, keep three replicated copies for 10 years (3,652 days), one in Site 1, one in Site 2, and one in Site 3. At the end of 10 years, delete all copies of the object.

ILM rule 2 for example 4: Save two copies of noncurrent versions for 2 years

This example ILM rule stores two copies of the noncurrent versions of an S3 versioned object for 2 years.

Because ILM rule 1 applies to all versions of the object, you must create another rule to filter out any noncurrent versions.

To create a rule that uses “Noncurrent time” as the reference time, select **Yes** for the question, “Apply this rule to older object versions only (in S3 buckets with versioning enabled)?” in Step 1 (Enter details) of the Create an ILM rule wizard. When you select **Yes**, *Noncurrent time* is automatically selected for the reference time, and you can’t select a different reference time.



In this example, only two copies of the noncurrent versions are stored, and those copies will be stored for two years.

Rule definition	Example value
Storage Pools	Two storage pools, each at different data centers, Site 1 and Site 2.
Rule name	Noncurrent Versions: Two Copies Two Years
Reference time	Noncurrent time Automatically selected when you select **Yes** for the question, “Apply this rule to older object versions only (in S3 buckets with versioning enabled)?” in the Create an ILM rule wizard.
Placements	On Day 0 relative to noncurrent time (that is, starting from the day the object version becomes the noncurrent version), keep two replicated copies of the noncurrent object versions for 2 years (730 days), one in Site 1 and one in Site 2. At the end of 2 years, delete the noncurrent versions.

ILM policy for example 4: S3 versioned objects

If you want to manage older versions of an object differently than the current version, rules that use “Noncurrent time” as the reference time must appear in the ILM policy before rules that apply to the current object version.

An ILM policy for S3 versioned objects might include ILM rules such as the following:

- Keep any older (noncurrent) versions of each object for 2 years, starting from the day the version became noncurrent.

> The “Noncurrent time” rules must appear in the policy before the rules that apply to the current object version. Otherwise, the noncurrent object versions will never be matched by the “Noncurrent time” rule.

- At ingest, create three replicated copies and store one copy at each of three sites. Keep copies of the current object version for 10 years.

When you simulate the example policy, you would expect test objects to be evaluated as follows:

- Any noncurrent object versions would be matched by the first rule. If a noncurrent object version is older than 2 years, it is permanently deleted by ILM (all copies of the noncurrent version removed from the grid).

> To simulate noncurrent object versions, you must use that version’s UUID or CBID. While the object is still current, you can use object metadata lookup to find its UUID and CBID.

- The current object version would be matched by the second rule. When the current object version has been stored for 10 years, the ILM process adds a delete marker as the current version of the object, and it makes the previous object version “noncurrent”. The next time ILM evaluation occurs, this noncurrent version is matched by the first rule. As a result, the copy at Site 3 is purged and the two copies at Site 1 and Site 2 are stored for 2 more years.

Example 5: ILM rules and policy for Strict ingest behavior

You can use a location filter and the Strict ingest behavior in a rule to prevent objects from being saved at a particular data center location.

In this example, a Paris-based tenant does not want to store some objects outside of the EU because of regulatory concerns. Other objects, including all objects from other tenant accounts, can be stored at either the Paris data center or the US data center.

> The following ILM rules and policy are only examples. There are many ways to configure ILM rules. Before activating a new policy, simulate the proposed policy to confirm it will work as intended to protect content from loss.

Related information

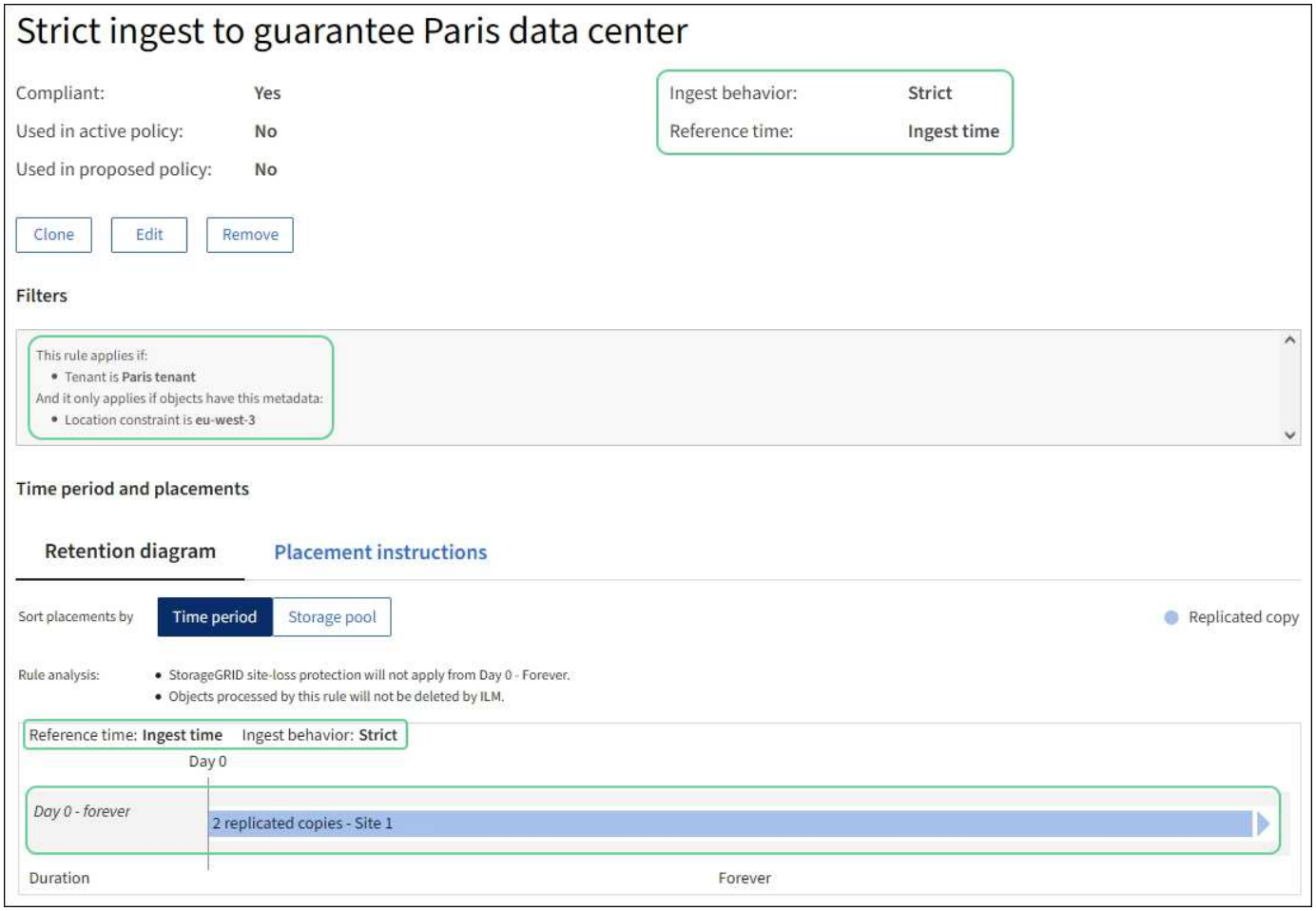
- Ingest options
- Create ILM rule: Select ingest behavior

ILM rule 1 for example 5: Strict ingest to guarantee Paris data center

This example ILM rule uses the Strict ingest behavior to guarantee that objects saved by a Paris-based tenant to S3 buckets with the region set to eu-west-3 region (Paris) are never stored at the US data center.

This rule applies to objects that belong to the Paris tenant and that have the S3 bucket region set to eu-west-3 (Paris).

Rule definition	Example value
Tenant account	Paris tenant
Advanced filter	Location constraint equals eu-west-3
Storage pools	Site 1 (Paris)
Rule name	Strict ingest to guarantee Paris data center
Reference time	Ingest time
Placements	On Day 0, keep two replicated copies forever in Site 1 (Paris)
Ingest behavior	Strict. Always use this rule's placements on ingest. Ingest fails if it is not possible to store two copies of the object at the Paris data center.



ILM rule 2 for example 5: Balanced ingest for other objects

This example ILM rule uses the Balanced ingest behavior to provide optimum ILM efficiency for any objects not matched by the first rule. Two copies of all objects matched by this rule will be stored—one at the US data center and one at the Paris data center. If the rule can't be satisfied immediately, interim copies are stored at any available location.

This rule applies to objects that belong to any tenant and any region.

Rule definition	Example value
Tenant account	Ignore
Advanced filter	*Not specified*
Storage pools	Site 1 (Paris) and Site 2 (US)
Rule name	2 Copies 2 Data Centers
Reference time	Ingest time
Placements	On Day 0, keep two replicated copies forever at two data centers

Rule definition	Example value
Ingest behavior	Balanced. Objects that match this rule are placed according to the rule's placement instructions if possible. Otherwise, interim copies are made at any available location.

ILM policy for example 5: Combining ingest behaviors

The example ILM policy includes two rules that have different ingest behaviors.

An ILM policy that uses two different ingest behaviors might include ILM rules such as the following:

- Store objects that belong to the Paris tenant and that have the S3 bucket region set to eu-west-3 (Paris) only in the Paris data center. Fail ingest if the Paris data center is not available.
- Store all other objects (including those that belong to the Paris tenant but that have a different bucket region) in both the US data center and the Paris data center. Make interim copies in any available location if the placement instruction can't be satisfied.

When you simulate the example policy, you expect test objects to be evaluated as follows:

- Any objects that belong to the Paris tenant and that have the S3 bucket region set to eu-west-3 are matched by the first rule and are stored at the Paris data center. Because the first rule uses Strict ingest, these objects are never stored at the US data center. If the Storage Nodes at the Paris data center aren't available, ingest fails.
- All other objects are matched by the second rule, including objects that belong to the Paris tenant and that don't have the S3 bucket region set to eu-west-3. One copy of each object is saved at each data center. However, because the second rule uses Balanced ingest, if one data center is unavailable, two interim copies are saved at any available location.

Example 6: Change an ILM policy

If your data protection needs to be changed or you add new sites, you can create and activate a new ILM policy.

Before changing a policy, you must understand how changes in ILM placements can temporarily affect the overall performance of a StorageGRID system.

In this example, a new StorageGRID site has been added in an expansion, and a new active ILM policy needs to be implemented to store data at the new site. To implement a new active policy, first create a proposed policy by either cloning an existing policy *or* starting from scratch. Afterward, you must simulate and then activate the new policy.

> The following ILM rules and policy are only examples. There are many ways to configure ILM rules. Before activating a new policy, simulate the proposed policy to confirm it will work as intended to protect content from loss.

How changing an ILM policy affects performance

When you activate a new ILM policy, the performance of your StorageGRID system might be temporarily affected, especially if the placement instructions in the new policy require many existing objects to be moved to new locations.

When you activate a new ILM policy, StorageGRID uses it to manage all objects, including existing objects and newly ingested objects. Before activating a new ILM policy, review any changes to the placement of existing replicated and erasure-coded objects. Changing an existing object's location might result in temporary resource issues when the new placements are evaluated and implemented.

To ensure a new ILM policy does not affect the placement of existing replicated and erasure-coded objects, you can create an ILM rule with an ingest time filter. For example, **Ingest time *is on or after <date and time>***, so that the new rule applies only to objects ingested on or after the date and time specified.

The types of ILM policy changes that can temporarily affect StorageGRID performance include the following:

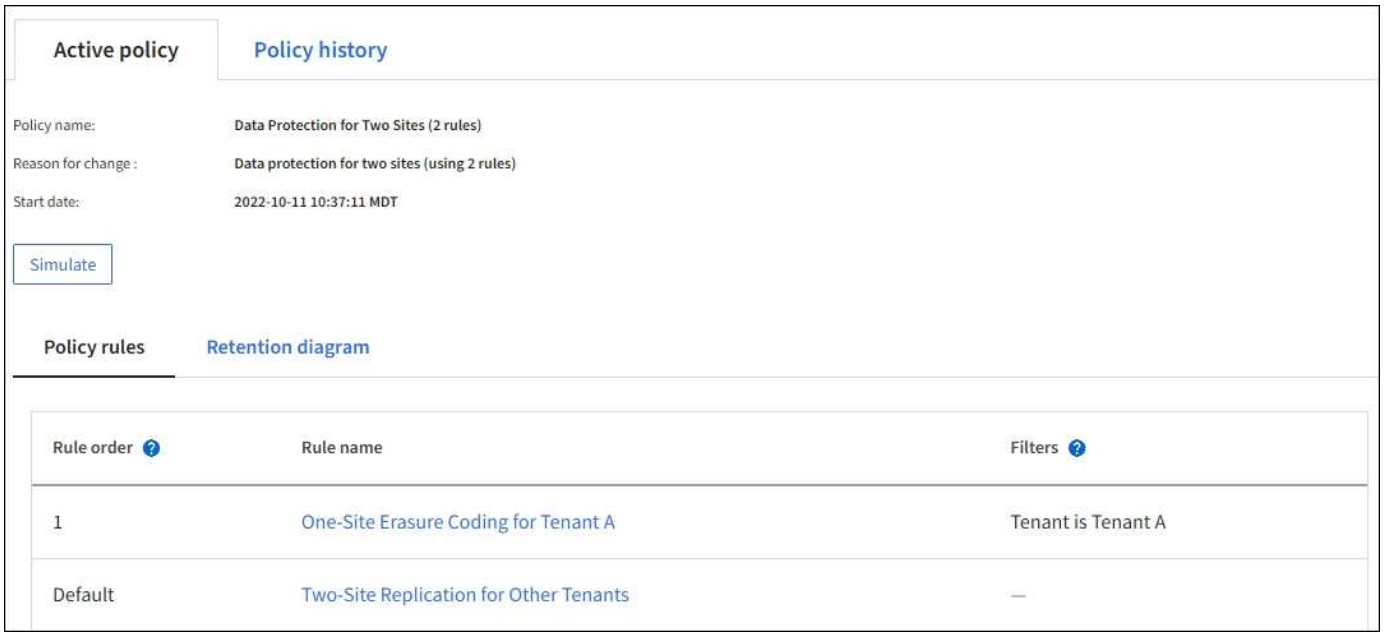
- Applying a different erasure coding profile to existing erasure-coded objects.

> StorageGRID considers each erasure coding profile to be unique and does not reuse erasure-coding fragments when a new profile is used.

- Changing the type of copies required for existing objects; for example, converting a large percentage of replicated objects to erasure-coded objects.
- Moving copies of existing objects to a completely different location; for example, moving a large number of objects to or from a Cloud Storage Pool or to or from a remote site.

Active ILM policy for example 6: Data protection at two sites

In this example, the active ILM policy was initially designed for a two-site StorageGRID system and uses two ILM rules.



In this ILM policy, objects belonging to Tenant A are protected by 2+1 erasure coding at a single site, while objects belonging to all other tenants are protected across two sites using 2-copy replication.

> The first rule in this example uses an advanced filter to ensure that erasure coding is not used for small objects. Any of Tenant A's objects that are smaller than 1 MB will be protected by the default rule, which uses replication.

Rule 1: One-site erasure coding for Tenant A

Rule definition	Example value
Rule name	One-Site Erasure Coding for Tenant A
Tenant Account	Tenant A
Storage Pool	Site 1
Placements	2+1 erasure coding in Site 1 from day 0 to forever

Rule 2: Two-site replication for other tenants

Rule definition	Example value
Rule name	Two-Site Replication for Other Tenants
Tenant Account	Ignore
Storage Pools	Site 1 and Site 2
Placements	Two replicated copies from day 0 to forever: one copy at Site 1 and one copy at Site 2.

Proposed ILM policy for example 6: Data protection at three sites

In this example, the ILM policy is being replaced with a new policy for a three-site StorageGRID system.

After performing an expansion to add the new site, the grid administrator created two new storage pools: a storage pool for Site 3 and a storage pool containing all three sites (not the same as the All Storage Nodes default storage pool). Then, the administrator created two new ILM rules and a new proposed ILM policy, which is designed to protect data at all three sites.

When this new ILM policy is activated, objects belonging to Tenant A will be protected by 2+1 erasure coding at three sites, while objects belonging to other tenants (and smaller objects belonging to Tenant A) will be protected across three sites using 3-copy replication.

Rule 1: Three-site erasure coding for Tenant A

Rule definition	Example value
Rule name	Three-Site Erasure Coding for Tenant A
Tenant Account	Tenant A
Storage Pool	All 3 Sites (includes Site 1, Site 2, and Site 3)

Rule definition	Example value
Placements	2+1 erasure coding in All 3 Sites from day 0 to forever

Rule 2: Three-site replication for other tenants

Rule definition	Example value
Rule name	Three-Site Replication for Other Tenants
Tenant Account	Ignore
Storage Pools	Site 1, Site 2, and Site 3
Placements	Three replicated copies from day 0 to forever: one copy at Site 1, one copy at Site 2, and one copy at Site 3.

Activating the proposed ILM policy for example 6

When you activate a new proposed ILM policy, existing objects might be moved to new locations or new object copies might be created for existing objects, based on the placement instructions in any new or updated rules.

Errors in an ILM policy can cause unrecoverable data loss. Carefully review and simulate the policy before activating it to confirm that it will work as intended.

When you activate a new ILM policy, StorageGRID uses it to manage all objects, including existing objects and newly ingested objects. Before activating a new ILM policy, review any changes to the placement of existing replicated and erasure-coded objects. Changing an existing object's location might result in temporary resource issues when the new placements are evaluated and implemented.

What happens when erasure-coding instructions change

In the currently active ILM policy for this example, objects belonging to Tenant A are protected using 2+1 erasure coding at Site 1. In the new proposed ILM policy, objects belonging to Tenant A will be protected using 2+1 erasure coding at Sites 1, 2, and 3.

When the new ILM policy is activated, the following ILM operations occur:

- New objects ingested by Tenant A are split into two data fragments and one parity fragment is added. Then, each of the three fragments is stored at a different site.
- The existing objects belonging to Tenant A are re-evaluated during the ongoing ILM scanning process. Because the ILM placement instructions use a new erasure coding profile, entirely new erasure-coded fragments are created and distributed to the three sites.

The existing 2+1 fragments at Site 1 aren't reused. StorageGRID considers each erasure coding profile to be unique and does not reuse erasure-coding fragments when a new profile is used.

What happens when replication instructions change

In the currently active ILM policy for this example, objects belonging other tenants are protected using two replicated copies in storage pools at Sites 1 and 2. In the new proposed ILM policy, objects belonging to other tenants will be protected using three replicated copies in storage pools at Sites 1, 2, and 3.

When the new ILM policy is activated, the following ILM operations occur:

- When any tenant other than Tenant A ingests a new object, StorageGRID creates three copies and saves one copy at each site.
- Existing objects belonging to these other tenants are re-evaluated during the ongoing ILM scanning process. Because the existing object copies at Site 1 and Site 2 continue to satisfy the replication requirements of the new ILM rule, StorageGRID only needs to create one new copy of the object for Site 3.

Performance impact of activating this policy

When the proposed ILM policy in this example is activated, the overall performance of this StorageGRID system will be temporarily affected. Higher than normal levels of grid resources will be required to create new erasure-coded fragments for Tenant A's existing objects and new replicated copies at Site 3 for other tenants' existing objects.

As a result of the ILM policy change, client read and write requests might temporarily experience higher than normal latencies. Latencies will return to normal levels after the placement instructions are fully implemented across the grid.

To avoid resource issues when activating a new ILM policy, you can use the Ingest time advanced filter in any rule that might change the location of large numbers of existing objects. Set Ingest time to be greater than or equal to the approximate time when the new policy will go into effect to ensure that existing objects aren't moved unnecessarily.

> Contact technical support if you need to slow or increase the rate at which objects are processed after an ILM policy change.

Example 7: Compliant ILM policy for S3 Object Lock

You can use the S3 bucket, ILM rules, and ILM policy in this example as a starting point when defining an ILM policy to meet the object protection and retention requirements for objects in buckets with S3 Object Lock enabled.

> If you used the legacy Compliance feature in previous StorageGRID releases, you can also use this example to help manage any existing buckets that have the legacy Compliance feature enabled.

> The following ILM rules and policy are only examples. There are many ways to configure ILM rules. Before activating a new policy, simulate the proposed policy to confirm it will work as intended to protect content from loss.

Related information

- Manage objects with S3 Object Lock
- Create an ILM policy

Bucket and objects for S3 Object Lock example

In this example, an S3 tenant account named Bank of ABC has used the Tenant Manager to create a bucket with S3 Object Lock enabled to store critical bank records.

Bucket definition	Example value
Tenant Account Name	Bank of ABC
Bucket Name	bank-records
Bucket Region	us-east-1 (default)

Each object and object version that is added to the bank-records bucket will use the following values for `retain-until-date` and `legal hold` settings.

Setting for each object	Example value
`retain-until-date`	`"2030-12-30T23:59:59Z"` (December 30, 2030) Each object version has its own `retain-until-date` setting. This setting can be increased, but not decreased.
`legal hold`	`"OFF"` (Not in effect) A legal hold can be placed or lifted on any object version at any time during the retention period. If an object is under a legal hold, the object can't be deleted even if the `retain-until-date` has been reached.

ILM rule 1 for S3 Object Lock example: Erasure coding profile with bucket matching

This example ILM rule applies only to the S3 tenant account named Bank of ABC. It matches any object in the `bank-records` bucket and then uses erasure coding to store the object on Storage Nodes at three data center sites using a 6+3 erasure coding profile. This rule satisfies the requirements of buckets with S3 Object Lock enabled: a copy is kept on Storage Nodes from day 0 to forever, using Ingest time as the reference time.

Rule definition	Example value
Rule name	Compliant Rule: EC Objects in bank-records Bucket - Bank of ABC
Tenant Account	Bank of ABC
Bucket Name	`bank-records`

Rule definition	Example value
Advanced filter	Object Size (MB) greater than 1 **Note:** This filter ensures that erasure coding is not used for objects 1 MB or smaller.

Rule definition	Example value
Reference time	Ingest time
Placements	From day 0 store forever
Erasure Coding Profile	• Create an erasure-coded copy on Storage Nodes at three data center sites • Uses 6+3 erasure-coding scheme

ILM rule 2 for S3 Object Lock example: Non-compliant rule

This example ILM rule initially stores two replicated object copies on Storage Nodes. After one year, it stores one copy on a Cloud Storage Pool forever. Because this rule uses a Cloud Storage Pool, it is not compliant and will not apply to the objects in buckets with S3 Object Lock enabled.

Rule definition	Example value
Rule name	Non-compliant rule: Use Cloud Storage Pool
Tenant accounts	Not specified
Bucket name	Not specified, but will only apply to buckets that don’t have S3 Object Lock (or the legacy Compliance feature) enabled.
Advanced filter	Not specified

Rule definition	Example value
Reference time	Ingest time
Placements	• On Day 0, keep two replicated copies on Storage Nodes in Data Center 1 and Data Center 2 for 365 days • After 1 year, keep one replicated copy in a Cloud Storage Pool forever

ILM rule 3 for S3 Object Lock example: Default rule

This example ILM rule copies object data to storage pools in two data centers. This compliant rule is designed to be the default rule in the ILM policy. It does not include any filters, does not use the Noncurrent reference

time, and satisfies the requirements of buckets with S3 Object Lock enabled: two object copies are kept on Storage Nodes from day 0 to forever, using Ingest as the reference time.

Rule definition	Example value
Rule name	Default compliant rule: Two Copies Two Data Centers
Tenant account	Not specified
Bucket name	Not specified
Advanced filter	Not specified

Rule definition	Example value
Reference time	Ingest time
Placements	From Day 0 to forever, keep two replicated copies—one on Storage Nodes in Data Center 1 and one on Storage Nodes in Data Center 2.

Compliant ILM policy for S3 Object Lock example

To create an ILM policy that will effectively protect all objects in your system, including those in buckets with S3 Object Lock enabled, you must select ILM rules that satisfy the storage requirements for all objects. Then, you must simulate and activate the proposed policy.

Add rules to the policy

In this example, the ILM policy includes three ILM rules, in the following order:

1. A compliant rule that uses erasure coding to protect objects greater than 1 MB in a specific bucket with S3 Object Lock enabled. The objects are stored on Storage Nodes from day 0 to forever.
2. A non-compliant rule that creates two replicated object copies on Storage Nodes for a year and then moves one object copy to a Cloud Storage Pool forever. This rule does not apply to buckets with S3 Object Lock enabled because it uses a Cloud Storage Pool.
3. The default compliant rule that creates two replicated object copies on Storage Nodes from day 0 to forever.

Simulate the proposed policy

After you have added rules in your proposed policy, chosen a default compliant rule, and arranged the other rules, you should simulate the policy by testing objects from the bucket with S3 Object Lock enabled and from other buckets. For example, when you simulate the example policy, you would expect test objects to be evaluated as follows:

- The first rule will only match test objects that are greater than 1 MB in the bucket bank-records for the Bank of ABC tenant.
- The second rule will match all objects in all non-compliant buckets for all other tenant accounts.
- The default rule will match these objects:

- Objects 1 MB or smaller in the bucket bank-records for the Bank of ABC tenant.
- Objects in any other bucket that has S3 Object Lock enabled for all other tenant accounts.

Activate the policy

When you are completely satisfied that the new policy protects object data as expected, you can activate it.

Copyright information

Trademark information