



# How objects are ingested

## StorageGRID

NetApp

October 03, 2025

# Table of Contents

- How objects are ingested . . . . . 1
  - Data-protection options for ingest . . . . . 1
    - Dual commit . . . . . 2
      - When to use the Dual commit option . . . . . 2
    - Strict . . . . . 2
      - When to use the Strict option . . . . . 2
    - Balanced . . . . . 2
      - When to use the Balanced option . . . . . 2
    - Flowchart of three ingest options . . . . . 2
- Advantages, disadvantages, and limitations of the data-protection options . . . . . 3
  - Advantages of the Balanced and Strict options . . . . . 3
  - Disadvantages of the Balanced and Strict options . . . . . 4
  - Limitations on object placements with the Balanced or Strict options . . . . . 5
  - How ILM rules and consistency controls interact to affect data protection . . . . . 5
  - Example of how the consistency control and ILM rule can interact . . . . . 5

# How objects are ingested

StorageGRID protects objects during ingest either by performing synchronous placement or by performing dual commit, as specified in the ILM rule that matches the objects.

When an S3 or Swift client stores an object to the grid, StorageGRID ingests the object using one of these two methods:

- **Synchronous placement:** StorageGRID immediately creates all object copies that are needed to meet ILM requirements. StorageGRID sends an “ingest successful” message to the client when all copies are created.

If StorageGRID cannot immediately create all object copies (for example, because a required location is temporarily unavailable), it either sends an “ingest failed” message to the client, or it falls back to creating interim object copies and evaluating ILM later, depending on the choice you made when you created the ILM rule.

- **Dual commit:** StorageGRID immediately creates two interim copies of the object, each on a different Storage Node, and sends an “ingest successful” message to the client. StorageGRID then queues the object for ILM evaluation.

When StorageGRID performs the ILM evaluation, it first checks to see if the interim copies satisfy the placement instructions in the ILM rule. For example, the two interim copies might satisfy the instructions in a two-copy ILM rule, but they would not satisfy the instructions in an erasure-coding rule. If the interim copies do not satisfy the ILM instructions, StorageGRID creates new object copies and deletes any interim copies that are not needed.

If StorageGRID cannot create two interim copies (for example, if a network issue prevents the second copy from being made), StorageGRID does not retry. Ingest fails.



S3 or Swift clients can specify that StorageGRID create a single interim copy at ingest by specifying `REDUCED_REDUNDANCY` for the storage class. See the instructions for implementing an S3 or Swift client for more information.

By default, StorageGRID uses synchronous placement to protect objects during ingest.

## Related information

[Data-protection options for ingest](#)

[Use S3](#)

[Use Swift](#)

## Data-protection options for ingest

When you create an ILM rule, you specify one of three options for protecting objects at ingest: Dual commit, Balanced, or Strict. Depending on your choice, StorageGRID makes interim copies and queues the objects for ILM evaluation later, or it uses synchronous placement and immediately makes copies to meet ILM requirements.

## Dual commit

When you select the Dual commit option, StorageGRID immediately makes interim object copies on two different Storage Nodes and returns an “ingest successful” message to the client. The object is queued for ILM evaluation, and copies that meet the rule’s placement instructions are made later.

### When to use the Dual commit option

Use the Dual commit option in either of these cases:

- You are using multi-site ILM rules and client ingest latency is your primary consideration. When using Dual commit, you must ensure your grid can perform the additional work of creating and removing the dual-commit copies if they do not satisfy ILM. Specifically:
  - The load on the grid must be low enough to prevent an ILM backlog.
  - The grid must have excess hardware resources (IOPS, CPU, memory, network bandwidth, and so on).
- You are using multi-site ILM rules and the WAN connection between the sites usually has high latency or limited bandwidth. In this scenario, using the Dual commit option can help prevent client timeouts. Before choosing the Dual commit option, you should test the client application with realistic workloads.

## Strict

When you select the Strict option, StorageGRID uses synchronous placement on ingest and immediately makes all object copies specified in the rule’s placement instructions. Ingest fails if StorageGRID cannot create all copies, for example, because a required storage location is temporarily unavailable. The client must retry the operation.

### When to use the Strict option

Use the Strict option if you have an operational or regulatory requirement to immediately store objects only in the locations outlined in the ILM rule. For example, to satisfy a regulatory requirement, you might need to use the Strict option and a Location Constraint advanced filter to guarantee that objects are never stored at certain data center.

[Example 5: ILM rules and policy for Strict ingest behavior](#)

## Balanced

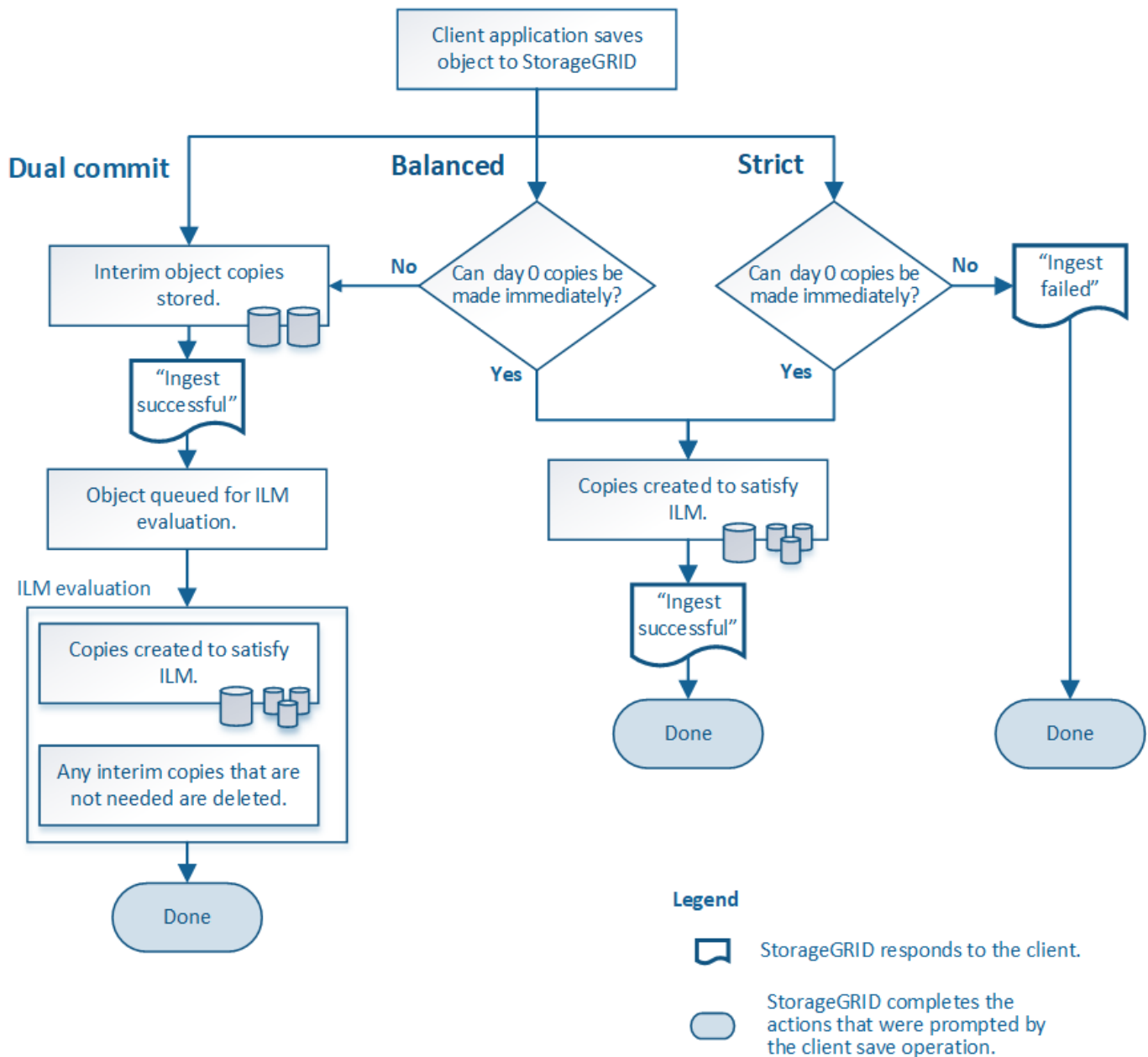
When you select the Balanced option, StorageGRID also uses synchronous placement on ingest and immediately makes all copies specified in the rule’s placement instructions. In contrast with the Strict option, if StorageGRID cannot immediately make all copies, it uses Dual commit instead.

### When to use the Balanced option

Use the Balanced option to achieve the best combination of data protection, grid performance, and ingest success. Balanced is the default option in the ILM rule wizard.

## Flowchart of three ingest options

The flowchart shows what happens when objects are matched by an ILM rule that uses one of these ingest options.



## Related information

[How objects are ingested](#)

# Advantages, disadvantages, and limitations of the data-protection options

Understanding the advantages and disadvantages of each of the three options for protecting data at ingest (Balanced, Strict, or Dual commit) can help you decide which one to select for an ILM rule.

## Advantages of the Balanced and Strict options

When compared to Dual commit, which creates interim copies during ingest, the two synchronous placement options can provide the following advantages:

- **Better data security:** Object data is immediately protected as specified in the ILM rule's placement instructions, which can be configured to protect against a wide variety of failure conditions, including the failure of more than one storage location. Dual commit can only protect against the loss of a single local copy.
- **More efficient grid operation:** Each object is processed only once, as it is ingested. Because the StorageGRID system does not need to track or delete interim copies, there is less processing load and less database space is consumed.
- **(Balanced) Recommended:** The Balanced option provides optimal ILM efficiency. Using the Balanced option is recommended unless Strict ingest behavior is required or the grid meets all of the criteria for using for Dual commit.
- **(Strict) Certainty about object locations:** The Strict option guarantees that objects are immediately stored according to the placement instructions in the ILM rule.

## Disadvantages of the Balanced and Strict options

When compared to Dual commit, the Balanced and Strict options have some disadvantages:

- **Longer client ingests:** Client ingest latencies might be longer. When you use the Balanced and Strict options, an "ingest successful" message is not returned to the client until all erasure-coded fragments or replicated copies are created and stored. However, object data will most likely reach its final placement much faster.
- **(Strict) Higher rates of ingest failure:** With the Strict option, ingest fails whenever StorageGRID cannot immediately make all copies specified in the ILM rule. You might see high rates of ingest failure if a required storage location is temporarily offline or if network issues cause delays in copying objects between sites.
- **(Strict) S3 multipart upload placements might not be as expected in some circumstances:** With Strict, you expect objects either to be placed as described by the ILM rule or for ingest to fail. However, with an S3 multipart upload, ILM is evaluated for each part of the object as it is ingested, and for the object as a whole when the multipart upload completes. In the following circumstances this might result in placements that are different than you expect:
  - **If ILM changes while an S3 multipart upload is in progress:** Because each part is placed according to the rule that is active when the part is ingested, some parts of the object might not meet current ILM requirements when the multipart upload completes. In these cases, ingest of the object does not fail. Instead, any part that is not placed correctly is queued for ILM re-evaluation, and is moved to the correct location later.
  - **When ILM rules filter on size:** When evaluating ILM for a part, StorageGRID filters on the size of the part, not the size of the object. This means that parts of an object can be stored in locations that do not meet ILM requirements for the object as a whole. For example, if a rule specifies that all objects 10 GB or larger are stored at DC1 while all smaller objects are stored at DC2, at ingest each 1 GB part of a 10-part multipart upload is stored at DC2. When ILM is evaluated for the object, all parts of the object are moved to DC1.
- **(Strict) Ingest does not fail when object tags or metadata are updated and newly required placements cannot be made:** With Strict, you expect objects either to be placed as described by the ILM rule or for ingest to fail. However, when you update metadata or tags for an object that is already stored in the grid, the object is not re-ingested. This means that any changes to object placement that are triggered by the update are not made immediately. Placement changes are made when ILM is re-evaluated by normal background ILM processes. If required placement changes cannot be made (for example, because a newly required location is unavailable), the updated object retains its current placement until the placement changes are possible.

## Limitations on object placements with the Balanced or Strict options

The Balanced or Strict options cannot be used for ILM rules that have any of these placement instructions:

- Placement in a Cloud Storage Pool at day 0.
- Placement in an Archive Node at day 0.
- Placements in a Cloud Storage Pool or an Archive Node when the rule has a User Defined Creation Time as its Reference Time.

These restrictions exist because StorageGRID cannot synchronously make copies to a Cloud Storage Pool or an Archive Node, and a User Defined Creation Time could resolve to the present.

## How ILM rules and consistency controls interact to affect data protection

Both your ILM rule and your choice of consistency control affect how objects are protected. These settings can interact.

For example, the ingest behavior selected for an ILM rule affects the initial placement of object copies, while the consistency control used when an object is stored affects the initial placement of object metadata. Because StorageGRID requires access to both an object's metadata and its data to fulfill client requests, selecting matching levels of protection for the consistency level and ingest behavior can provide better initial data protection and more predictable system responses.

Here is a brief summary of the consistency controls that are available in StorageGRID:

- **all**: All nodes receive object metadata immediately or the request will fail.
- **strong-global**: Object metadata is immediately distributed to all sites. Guarantees read-after-write consistency for all client requests across all sites.
- **strong-site**: Object metadata is immediately distributed to other nodes at the site. Guarantees read-after-write consistency for all client requests within a site.
- **read-after-new-write**: Provides read-after-write consistency for new objects and eventual consistency for object updates. Offers high availability and data protection guarantees.
- **available** (eventual consistency for HEAD operations): Behaves the same as the “read-after-new-write” consistency level, but only provides eventual consistency for HEAD operations.



Before selecting a consistency level, read the full description of these settings in the instructions for creating an S3 or Swift client application. You should understand the benefits and limitations before changing the default value.

## Example of how the consistency control and ILM rule can interact

Suppose you have a two-site grid with the following ILM rule and the following consistency level setting:

- **ILM rule**: Create two object copies, one at the local site and one at a remote site. The Strict ingest behavior is selected.
- **Consistency level**: “strong-global” (Object metadata is immediately distributed to all sites.)

When a client stores an object to the grid, StorageGRID makes both object copies and distributes metadata to both sites before returning success to the client.

The object is fully protected against loss at the time of the ingest successful message. For example, if the local

site is lost shortly after ingest, copies of both the object data and the object metadata still exist at the remote site. The object is fully retrievable.

If you instead used the same ILM rule and the “strong-site” consistency level, the client might receive a success message after object data is replicated to the remote site but before object metadata is distributed there. In this case, the level of protection of object metadata does not match the level of protection for object data. If the local site is lost shortly after ingest, object metadata is lost. The object cannot be retrieved.

The inter-relationship between consistency levels and ILM rules can be complex. Contact NetApp if you require assistance.

**Related information**

[What replication is](#)

[What erasure coding is](#)

[What erasure-coding schemes are](#)

[Example 5: ILM rules and policy for Strict ingest behavior](#)

[Use S3](#)

[Use Swift](#)

## Copyright information

Copyright © 2025 NetApp, Inc. All Rights Reserved. Printed in the U.S. No part of this document covered by copyright may be reproduced in any form or by any means—graphic, electronic, or mechanical, including photocopying, recording, taping, or storage in an electronic retrieval system—without prior written permission of the copyright owner.

Software derived from copyrighted NetApp material is subject to the following license and disclaimer:

THIS SOFTWARE IS PROVIDED BY NETAPP “AS IS” AND WITHOUT ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WHICH ARE HEREBY DISCLAIMED. IN NO EVENT SHALL NETAPP BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

NetApp reserves the right to change any products described herein at any time, and without notice. NetApp assumes no responsibility or liability arising from the use of products described herein, except as expressly agreed to in writing by NetApp. The use or purchase of this product does not convey a license under any patent rights, trademark rights, or any other intellectual property rights of NetApp.

The product described in this manual may be protected by one or more U.S. patents, foreign patents, or pending applications.

LIMITED RIGHTS LEGEND: Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (b)(3) of the Rights in Technical Data -Noncommercial Items at DFARS 252.227-7013 (FEB 2014) and FAR 52.227-19 (DEC 2007).

Data contained herein pertains to a commercial product and/or commercial service (as defined in FAR 2.101) and is proprietary to NetApp, Inc. All NetApp technical data and computer software provided under this Agreement is commercial in nature and developed solely at private expense. The U.S. Government has a non-exclusive, non-transferrable, nonsublicensable, worldwide, limited irrevocable license to use the Data only in connection with and in support of the U.S. Government contract under which the Data was delivered. Except as provided herein, the Data may not be used, disclosed, reproduced, modified, performed, or displayed without the prior written approval of NetApp, Inc. United States Government license rights for the Department of Defense are limited to those rights identified in DFARS clause 252.227-7015(b) (FEB 2014).

## Trademark information

NETAPP, the NETAPP logo, and the marks listed at <http://www.netapp.com/TM> are trademarks of NetApp, Inc. Other company and product names may be trademarks of their respective owners.