



# **ILM y ciclo de vida de los objetos**

## StorageGRID

NetApp  
November 04, 2025

# Tabla de contenidos

ILM y ciclo de vida de los objetos . . . . .	1
Cómo funciona ILM a lo largo de la vida de un objeto . . . . .	1
Cómo se ingieren los objetos . . . . .	2
Opciones de procesamiento . . . . .	2
Ventajas, desventajas y limitaciones de las opciones de ingestión . . . . .	4
Cómo se almacenan los objetos (codificación de borrado o replicación) . . . . .	7
¿Qué es la replicación? . . . . .	7
Por qué no se debe utilizar la replicación de copia única . . . . .	8
¿Qué es el código de borrado? . . . . .	11
¿Qué son los esquemas de código de borrado? . . . . .	13
Ventajas, desventajas y requisitos de codificación de borrado . . . . .	16
Cómo se determina la retención de objetos . . . . .	18
Cómo los usuarios de inquilinos controlan la retención de objetos . . . . .	18
Cómo los administradores de grid controlan la retención de objetos . . . . .	18
Cómo interaccionan el ciclo de vida de bloque y ILM de S3 . . . . .	19
Ejemplos para la retención de objetos . . . . .	19
Cómo se eliminan los objetos . . . . .	20
Tiempo necesario para eliminar objetos . . . . .	21
Cómo se eliminan los objetos con versiones de S3 . . . . .	22

# ILM y ciclo de vida de los objetos

## Cómo funciona ILM a lo largo de la vida de un objeto

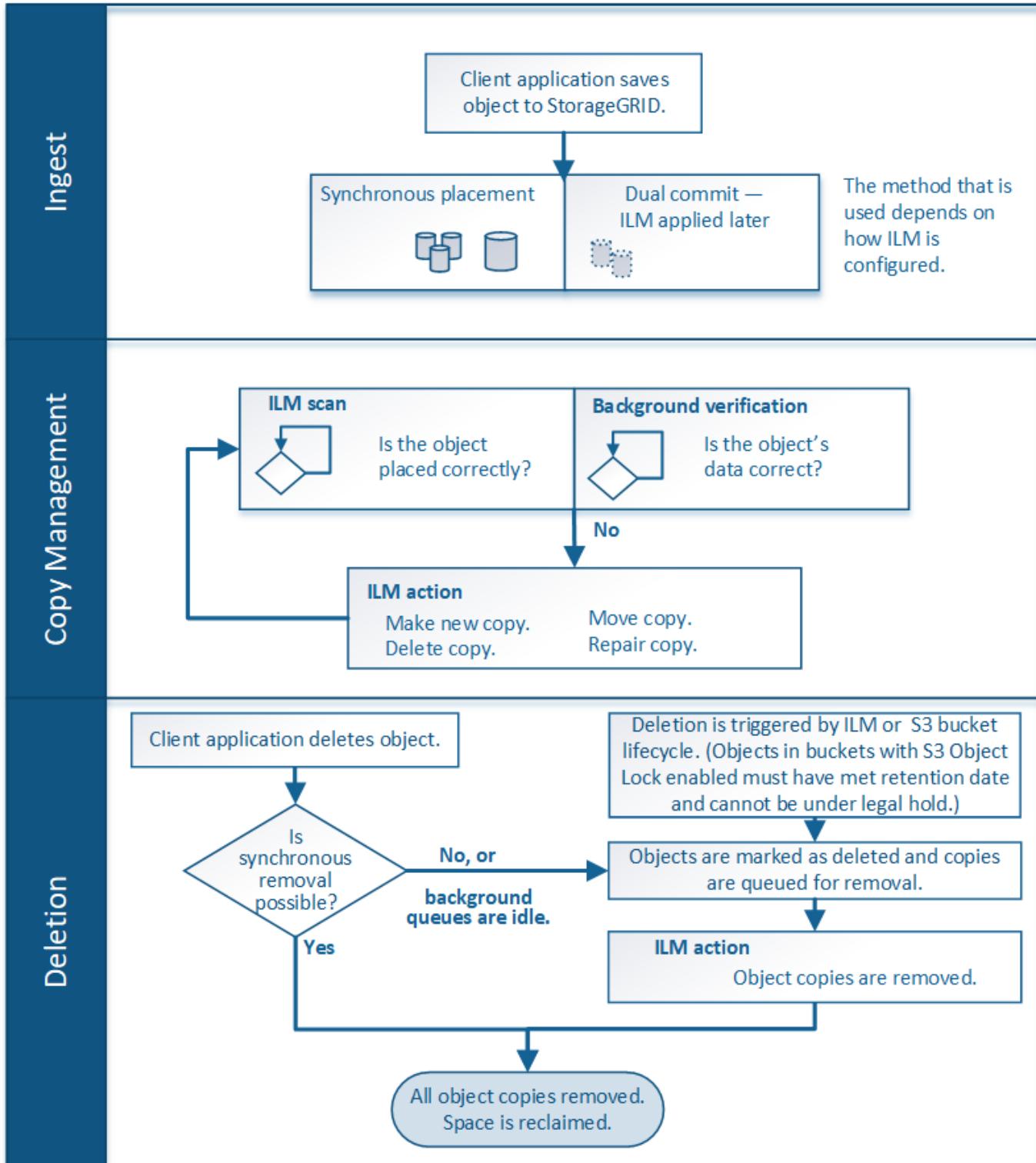
Comprender cómo utiliza StorageGRID ILM para gestionar objetos durante cada fase de su vida útil puede ayudarle a diseñar una política más eficaz.

- **Ingesta:** La ingestra comienza cuando una aplicación cliente S3 o Swift establece una conexión para guardar un objeto en el sistema StorageGRID, y se completa cuando StorageGRID devuelve un mensaje "ingesta correcta" al cliente. Los datos de objetos se protegen durante la ingestra aplicando instrucciones de ILM inmediatamente (ubicación síncrona) o creando copias provisionales y aplicando ILM más tarde (registro doble), según cómo se especifiquen los requisitos de ILM.
- **Administración de copias:** Despues de crear el número y el tipo de copias de objetos que se especifican en las instrucciones de colocación de ILM, StorageGRID administra las ubicaciones de objetos y protege los objetos contra pérdidas.
  - Análisis y evaluación de ILM: StorageGRID analiza continuamente la lista de objetos almacenados en la cuadrícula y comprueba si las copias actuales cumplen los requisitos de ILM. Cuando se requieren diferentes tipos, números o ubicaciones de copias de objetos, StorageGRID crea, elimina o mueve copias según sea necesario.
  - Verificación en segundo plano: StorageGRID realiza de forma continua verificación en segundo plano para comprobar la integridad de los datos de objetos. Si se encuentra un problema, StorageGRID crea automáticamente una copia de objeto nueva o un fragmento de objeto con código de borrado de reemplazo en una ubicación que cumple los requisitos actuales de ILM. Consulte "["Verifique la integridad del objeto"](#)".
- **Eliminación de objetos:** La gestión de un objeto finaliza cuando se eliminan todas las copias del sistema StorageGRID. Los objetos se pueden eliminar como resultado de una solicitud de eliminación por parte de un cliente, o bien como resultado de la eliminación por ILM o la eliminación provocada por el vencimiento del ciclo de vida de un bloque de S3.



Los objetos de un depósito que tiene S3 Object Lock activado no se pueden eliminar si están en una retención legal o si se ha especificado una fecha de retención hasta que no se cumple.

El diagrama resume el funcionamiento de ILM a lo largo del ciclo de vida de un objeto.



## Cómo se ingieren los objetos

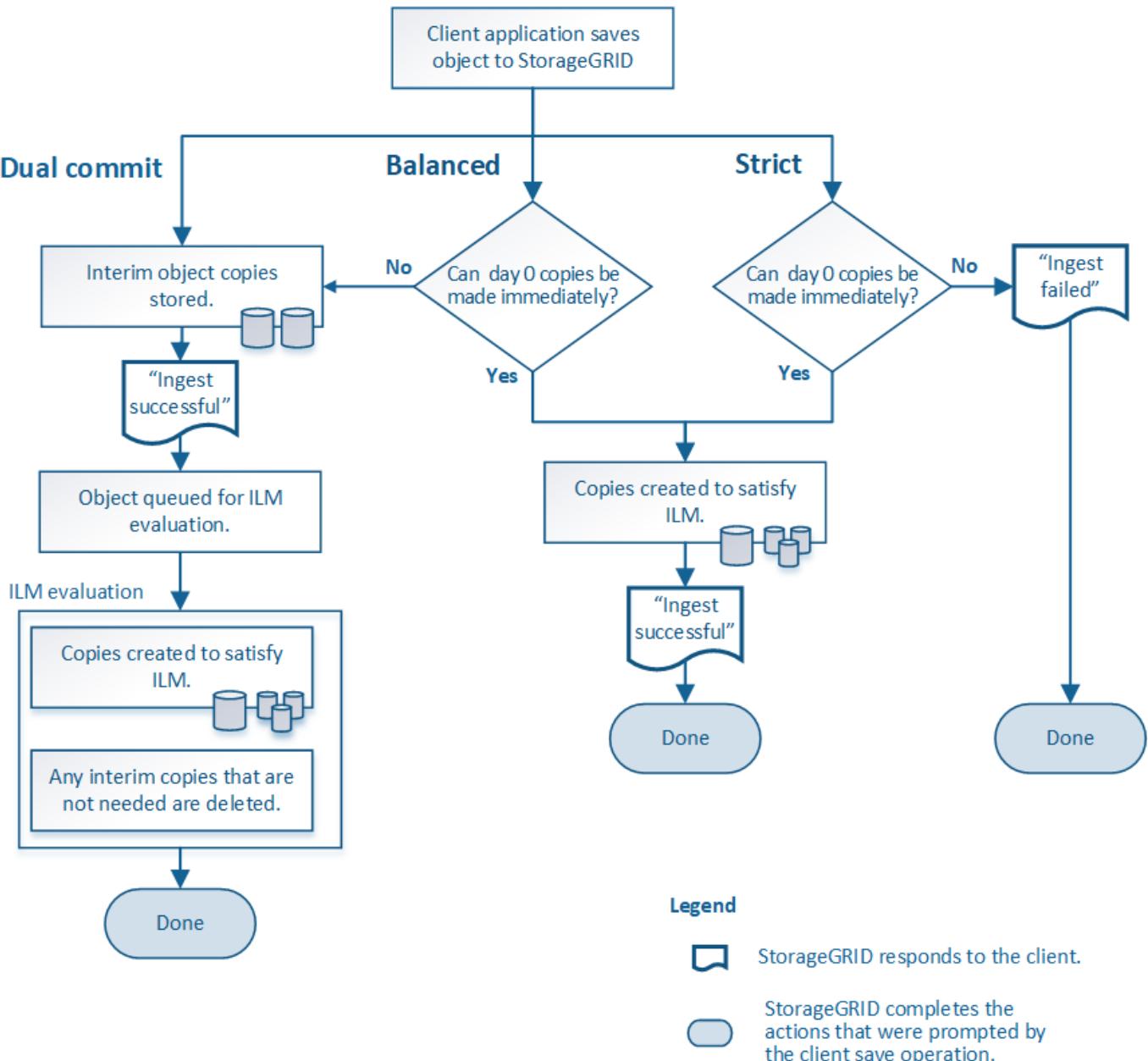
### Opciones de procesamiento

Al crear una regla de ILM, se deben especificar una de estas tres opciones para proteger los objetos durante la ingestión: Dual commit, strict o balanced.

Según elija, StorageGRID realiza copias provisionales y pone en cola los objetos para la evaluación de ILM más tarde, o utiliza una ubicación síncrona y realiza copias inmediatamente para cumplir los requisitos de ILM.

### Diagrama de flujo de opciones de ingestión

El diagrama de flujo muestra lo que ocurre cuando una regla de ILM se equipara con objetos que utilizan cada una de las tres opciones de ingestión.



### Registro doble

Al seleccionar la opción de confirmación doble, StorageGRID realiza inmediatamente copias provisionales de objetos en dos nodos de almacenamiento diferentes y devuelve un mensaje «'ingesta correcta'» al cliente. El objeto se pone en cola para la evaluación de ILM, y se realicen copias que cumplan con las instrucciones de ubicación de la regla más adelante.

## Cuándo utilizar la opción Dual COMMIT

Utilice la opción Dual Commit en uno de los siguientes casos:

- Está usando reglas de la ILM de varios sitios y la latencia de procesamiento de clientes es su principal consideración. Al utilizar la confirmación doble, debe asegurarse de que el grid pueda realizar el trabajo adicional de creación y eliminación de las copias de registro doble si no cumplen con el ciclo de vida de la información. Específicamente:
  - La carga en la cuadrícula debe ser lo suficientemente baja para evitar que se produzca una acumulación de ILM.
  - El grid debe tener un exceso de recursos de hardware (IOPS, CPU, memoria, ancho de banda de red, etc.).
- Utiliza reglas de ILM de varios sitios y la conexión WAN entre los sitios suele tener una alta latencia o un ancho de banda limitado. En este escenario, el uso de la opción Dual commit puede ayudar a evitar los tiempos de espera de los clientes. Antes de elegir la opción Dual commit, debe probar la aplicación cliente con cargas de trabajo realistas.

## Estricto

Al seleccionar la opción estricta, StorageGRID utiliza una ubicación síncrona al procesar y crea inmediatamente todas las copias de los objetos especificadas en las instrucciones de ubicación de la regla. La ingesta genera un error si StorageGRID no puede crear todas las copias, por ejemplo, porque no está disponible en estos momentos una ubicación de almacenamiento necesaria. El cliente debe volver a intentar la operación.

### Cuándo usar la opción estricta

Utilice la opción estricta si tiene un requisito operativo y de normativa para almacenar inmediatamente objetos solo en las ubicaciones descritas en la regla de ILM. Por ejemplo, para satisfacer un requisito normativo, es posible que necesite utilizar la opción estricta y un filtro avanzado de restricción de ubicación para garantizar que los objetos nunca se almacenen en determinados centros de datos.

Consulte "[Ejemplo 5: Reglas de ILM y política para el comportamiento de consumo estricto](#)".

## Equilibrada (predeterminado)

Cuando selecciona la opción equilibrada, StorageGRID también utiliza la ubicación síncrona durante la ingesta y hace inmediatamente todas las copias especificadas en las instrucciones de ubicación de la regla. En contraste con la opción estricta, si StorageGRID no puede hacer todas las copias de inmediato, utiliza Confirmación doble en su lugar.

### Cuándo utilizar la opción de equilibrio

Utilice la opción equilibrada para lograr la mejor combinación de protección de datos, rendimiento de grid y éxito de procesamiento. Equilibrada es la opción predeterminada del asistente Create ILM Rule.

## Ventajas, desventajas y limitaciones de las opciones de ingestión

Comprender las ventajas y las desventajas de cada una de las tres opciones de protección de datos en el procesamiento (confirmación equilibrada, estricta o doble) puede ayudarle a decidir cuál seleccionar para una regla de ILM.

Para obtener una descripción general de las opciones de ingestión, consulte "[Opciones de procesamiento](#)".

## Ventajas de las opciones equilibradas y estrictas

En comparación con el registro doble, que crea copias provisionales durante la ingestión, las dos opciones de colocación sincrónica pueden proporcionar las siguientes ventajas:

- **Mejor seguridad de datos:** Los datos de objeto están protegidos inmediatamente como se especifica en las instrucciones de colocación de la regla ILM, que se pueden configurar para proteger contra una amplia variedad de condiciones de fallo, incluyendo la falla de más de una ubicación de almacenamiento. La confirmación doble solo puede protegerse contra la pérdida de una única copia local.
- **Funcionamiento de red más eficiente:** Cada objeto se procesa una sola vez, ya que se ingiere. Dado que el sistema StorageGRID no necesita realizar un seguimiento o eliminar copias provisionales, hay menos carga de procesamiento y se consume menos espacio de la base de datos.
- **(equilibrado) recomendado:** La opción equilibrada proporciona una eficiencia óptima de ILM. Se recomienda utilizar la opción Balanced a menos que se requiera un comportamiento de ingestión estricto o que el grid cumpla todos los criterios para utilizar Dual commit.
- **(estricta) certeza acerca de las ubicaciones de objetos:** La opción estricta garantiza que los objetos se almacenen inmediatamente de acuerdo con las instrucciones de colocación en la regla ILM.

## Desventajas de las opciones equilibradas y estrictas

En comparación con la confirmación doble, las opciones equilibradas y estrictas tienen algunas desventajas:

- **Procesamiento de clientes más largos:** Las latencias de procesamiento de clientes pueden ser más largas. Cuando se utilizan las opciones equilibradas o estrictas, no se devuelve al cliente un mensaje «ingierir correctamente» hasta que se creen y almacenen todos los fragmentos con código de borrado o las copias replicadas. Sin embargo, lo más probable es que los datos de objetos lleguen a su ubicación final mucho más rápido.
- **(Estrictas) mayores tasas de fallo de ingestión:** Con la opción estricta, la ingestión falla siempre que StorageGRID no puede hacer inmediatamente todas las copias especificadas en la regla de ILM. Es posible que observe tasas elevadas de error de procesamiento si una ubicación de almacenamiento necesaria está temporalmente sin conexión o si los problemas de red provocan retrasos en la copia de objetos entre sitios.
- \* **(Estricta) las ubicaciones de carga de varias partes de S3 pueden no ser las esperadas en algunas circunstancias\*:** Con estricta, se espera que los objetos se coloquen como se describe en la regla ILM o que falle el procesamiento. Sin embargo, con una carga de varias partes de S3 KB, se evalúa ILM para cada parte del objeto conforme se procesa, y para el objeto en su conjunto cuando se completa la carga de varias partes. En las siguientes circunstancias, esto podría dar lugar a colocaciones que son diferentes de lo esperado:
  - **Si ILM cambia mientras una carga multipart de S3 está en curso:** Debido a que cada pieza se coloca según la regla que está activa cuando se ingiere la pieza, es posible que algunas partes del objeto no cumplan los requisitos actuales de ILM cuando se completa la carga de varias partes. En estos casos, la ingestión del objeto no falla. En su lugar, cualquier pieza que no se coloque correctamente se pone en cola para la reevaluación de ILM y posteriormente se mueve a la ubicación correcta.
  - **Cuando las reglas de ILM filtran el tamaño:** Al evaluar ILM para una pieza, StorageGRID filtra el tamaño de la pieza, no el tamaño del objeto. Esto significa que las partes de un objeto se pueden almacenar en ubicaciones que no cumplen con los requisitos de ILM para el objeto como un todo. Por ejemplo, si una regla especifica que todos los objetos de 10 GB o más se almacenan en DC1 mientras que todos los objetos más pequeños se almacenan en DC2, al ingerir cada parte de 1 GB de una carga multipart de 10 partes se almacena en DC2. Cuando se evalúa ILM para el objeto, todas las partes del objeto se mueven a DC1.

- **(estricta) la ingestión no falla cuando las etiquetas de objeto o los metadatos se actualizan y las colocaciones recientemente requeridas no se pueden hacer:** Con estricto, se espera que los objetos se coloquen como se describe en la regla ILM o que falle el procesamiento. Sin embargo, cuando se actualizan metadatos o etiquetas de un objeto que ya está almacenado en la cuadrícula, el objeto no se vuelve a procesar. Esto significa que cualquier cambio en la ubicación del objeto que se desencadene por la actualización no se realiza inmediatamente. Los cambios de colocación se realizan cuando la ILM se vuelve a evaluar por los procesos normales de ILM en segundo plano. Si no se pueden realizar cambios de ubicación necesarios (por ejemplo, porque no está disponible una nueva ubicación requerida), el objeto actualizado conserva su ubicación actual hasta que los cambios de ubicación sean posibles.

### **Limitaciones en la ubicación de objetos con las opciones equilibradas y estrictas**

Las opciones equilibradas o estrictas no se pueden usar para reglas de ILM que tengan alguna de estas instrucciones de ubicación:

- Ubicación en un pool de almacenamiento en cloud desde el día 0.
- Ubicación en un nodo de archivado en el día 0.
- Colocaciones en un pool de almacenamiento en la nube o en un nodo de archivado cuando la regla tiene un tiempo de creación definido por el usuario como su tiempo de referencia.

Estas restricciones se deben a que StorageGRID no puede realizar copias de forma síncrona en un grupo de almacenamiento en la nube o en un nodo de archivado, y una hora de creación definida por el usuario podría resolverse en el presente.

### **Cómo interactúan las reglas de ILM y los controles de coherencia para afectar a la protección de los datos**

Tanto la regla de ILM como la elección del control de coherencia afectan a la forma en que se protegen los objetos. Estos ajustes pueden interactuar.

Por ejemplo, el comportamiento de ingestión seleccionado para una regla de ILM afecta la colocación inicial de las copias de objetos, mientras que el control de consistencia utilizado cuando se almacena un objeto afecta la colocación inicial de los metadatos de objetos. Dado que StorageGRID requiere acceso a los datos de un objeto y los metadatos para satisfacer las solicitudes del cliente, la selección de niveles de protección correspondientes para el nivel de coherencia y el comportamiento de ingestión puede proporcionar mejor protección de datos inicial y respuestas del sistema más predecibles.

A continuación encontrará un breve resumen de los controles de consistencia disponibles en StorageGRID:

- **All:** Todos los nodos reciben metadatos de objeto inmediatamente o la solicitud falla.
- **Strong-global:** Los metadatos de objetos se distribuyen inmediatamente a todos los sitios. Garantiza la coherencia de lectura tras escritura para todas las solicitudes del cliente en todos los sitios.
- **Strong-site:** Los metadatos del objeto se distribuyen inmediatamente a otros nodos en el sitio. Garantiza la coherencia de lectura tras escritura para todas las solicitudes del cliente dentro de un sitio.
- **Read-after-new-write:** Proporciona consistencia de lectura-after-write para nuevos objetos y eventual consistencia para actualizaciones de objetos. Ofrece garantías de alta disponibilidad y protección de datos. Recomendado para la mayoría de los casos.
- **Disponible:** Proporciona consistencia eventual tanto para nuevos objetos como para actualizaciones de objetos. Para los cubos S3, utilice solo según sea necesario (por ejemplo, para un depósito que contiene valores de registro que rara vez se leen, o para operaciones HEAD u GET en claves que no existen). No se admite para bloques de FabricPool S3.



Antes de seleccionar un nivel de coherencia, lea la descripción completa de los controles de coherencia en las instrucciones para "["USE LA API DE REST DE S3"](#)". Debe comprender los beneficios y las limitaciones antes de cambiar el valor predeterminado.

### Ejemplo de cómo puede interactuar el control de consistencia y la regla de ILM

Suponga que tiene una cuadrícula de dos sitios con la siguiente regla de ILM y la siguiente configuración de nivel de coherencia:

- **Norma ILM:** Cree dos copias de objetos, una en el sitio local y otra en un sitio remoto. Se ha seleccionado el comportamiento de procesamiento estricto.
- **Nivel de coherencia:** "Strong-global" (los metadatos de objetos se distribuyen inmediatamente a todos los sitios).

Cuando un cliente almacena un objeto en el grid, StorageGRID realiza copias de objetos y distribuye los metadatos en ambos sitios antes de devolver el éxito al cliente.

El objeto está completamente protegido contra la pérdida en el momento del mensaje de procesamiento correcto. Por ejemplo, si el sitio local se pierde poco después del procesamiento, seguirán existiendo copias de los datos del objeto y los metadatos del objeto en el sitio remoto. El objeto se puede recuperar completamente.

Si en su lugar usa la misma regla de ILM y el nivel de consistencia de «*strong-site*», es posible que el cliente reciba un mensaje de éxito después de replicar los datos del objeto en el sitio remoto, pero antes de que los metadatos del objeto se distribuyan allí. En este caso, el nivel de protección de los metadatos de objetos no coincide con el nivel de protección de los datos de objetos. Si el sitio local se pierde poco después del procesamiento, se pierden los metadatos del objeto. No se puede recuperar el objeto.

La interrelación entre los niveles de coherencia y las reglas del ILM puede ser compleja. Póngase en contacto con NetApp si necesita ayuda.

#### Información relacionada

- ["Ejemplo 5: Reglas de ILM y política para el comportamiento de consumo estricto"](#)

## Cómo se almacenan los objetos (codificación de borrado o replicación)

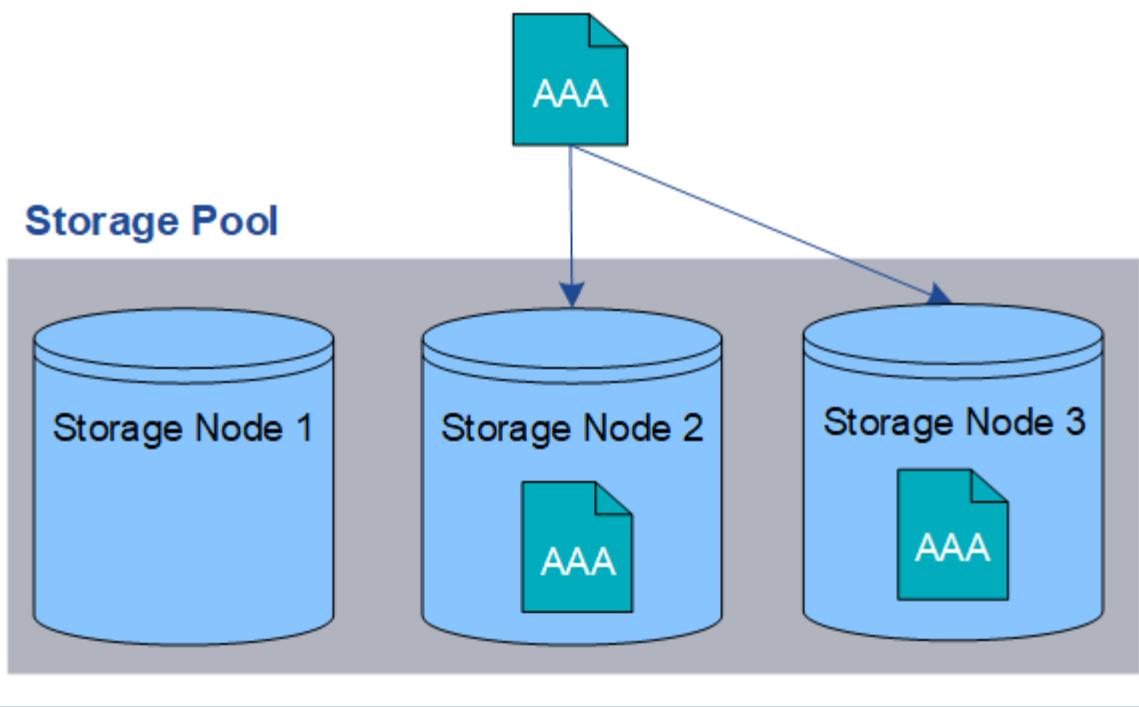
### ¿Qué es la replicación?

La replicación es uno de los dos métodos que utiliza StorageGRID para almacenar datos de objetos. Cuando los objetos coinciden con una regla de ILM que usa la replicación, el sistema crea copias exactas de datos de objetos y almacena las copias en nodos de almacenamiento o nodos de archivado.

Cuando configura una regla de ILM para crear copias replicadas, especifica cuántas copias se deben crear, dónde deben ubicarse y cuánto tiempo deben almacenarse las copias en cada ubicación.

En el ejemplo siguiente, la regla de ILM especifica que dos copias replicadas de cada objeto se coloquen en un pool de almacenamiento que contenga tres nodos de almacenamiento.

## Make 2 Copies



Cuando StorageGRID coincide con los objetos de esta regla, crea dos copias del objeto, colocando cada copia en un nodo de almacenamiento diferente en el pool de almacenamiento. Las dos copias pueden colocarse en dos de los tres nodos de almacenamiento disponibles. En este caso, la regla colocó copias de objetos en los nodos de almacenamiento 2 y 3. Debido a que hay dos copias, el objeto se puede recuperar si alguno de los nodos del pool de almacenamiento falla.

**i** StorageGRID solo puede almacenar una copia replicada de un objeto en un nodo de almacenamiento dado. Si el grid incluye tres nodos de almacenamiento y se crea una regla de gestión del ciclo de vida de la información de 4 copias, solo se crearán tres copias: Una por cada nodo de almacenamiento. Se activa la alerta **colocación de ILM inalcanzable** para indicar que la regla ILM no se pudo aplicar completamente.

### Información relacionada

- "[¿Qué es el código de borrado?](#)"
- "[¿Qué es un pool de almacenamiento?](#)"
- "[Habilite la protección contra pérdida de sitios mediante replicación y código de borrado](#)"

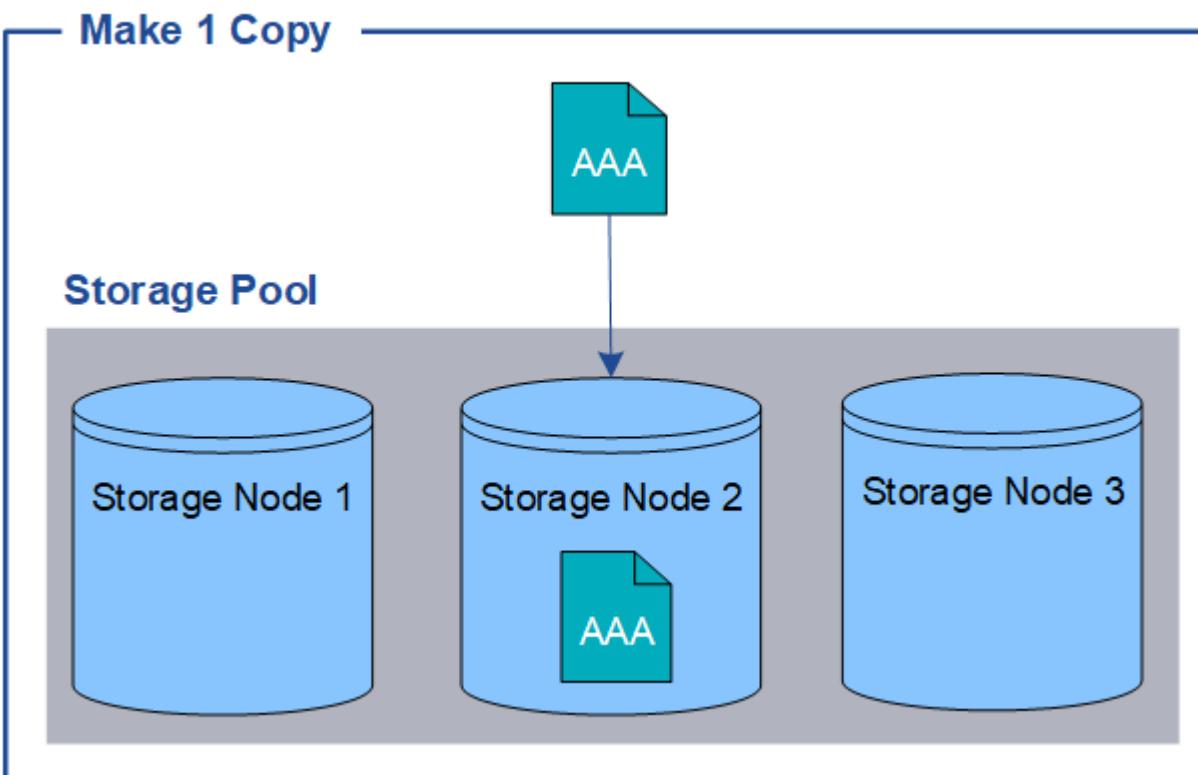
### Por qué no se debe utilizar la replicación de copia única

Al crear una regla de ILM para crear copias replicadas, debe especificar siempre al menos dos copias durante cualquier periodo de tiempo en las instrucciones de ubicación.

**i** No utilice una regla de ILM que cree solo una copia replicada durante un período de tiempo. Si sólo existe una copia replicada de un objeto, éste se pierde si falla un nodo de almacenamiento o tiene un error importante. También perderá temporalmente el acceso al objeto durante procedimientos de mantenimiento, como las actualizaciones.

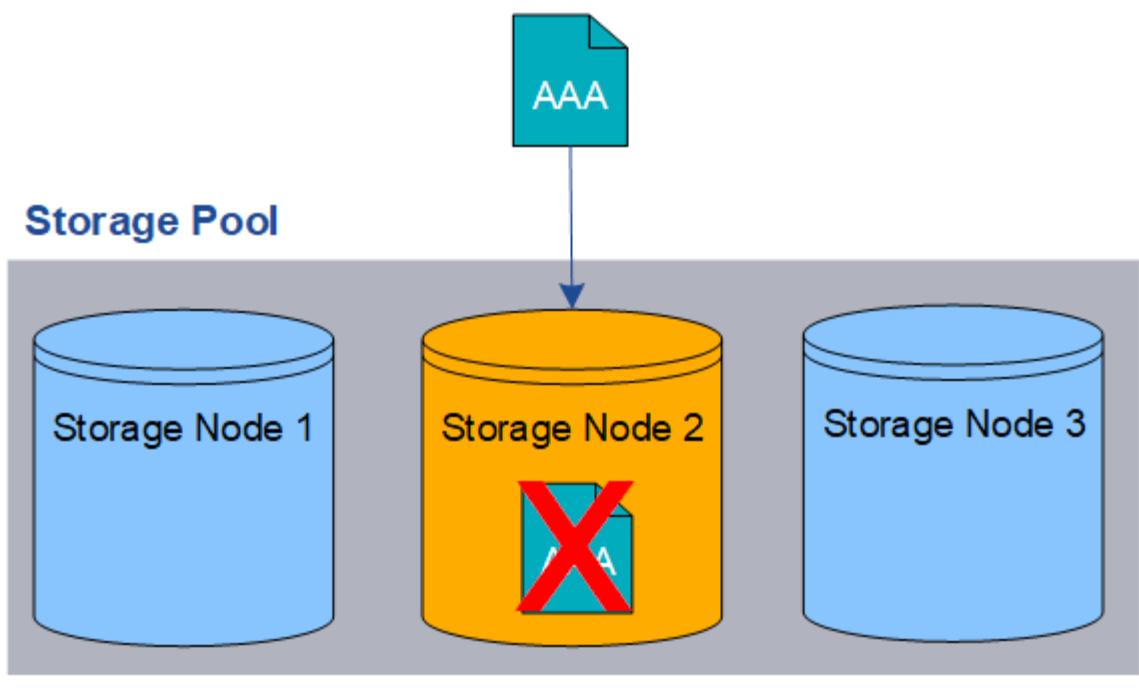
En el ejemplo siguiente, la regla Make 1 Copy ILM especifica que una copia replicada de un objeto se coloca

en un pool de almacenamiento que contiene tres nodos de almacenamiento. Cuando se ingiere un objeto que coincide con esta regla, StorageGRID coloca una sola copia en un solo nodo de almacenamiento.



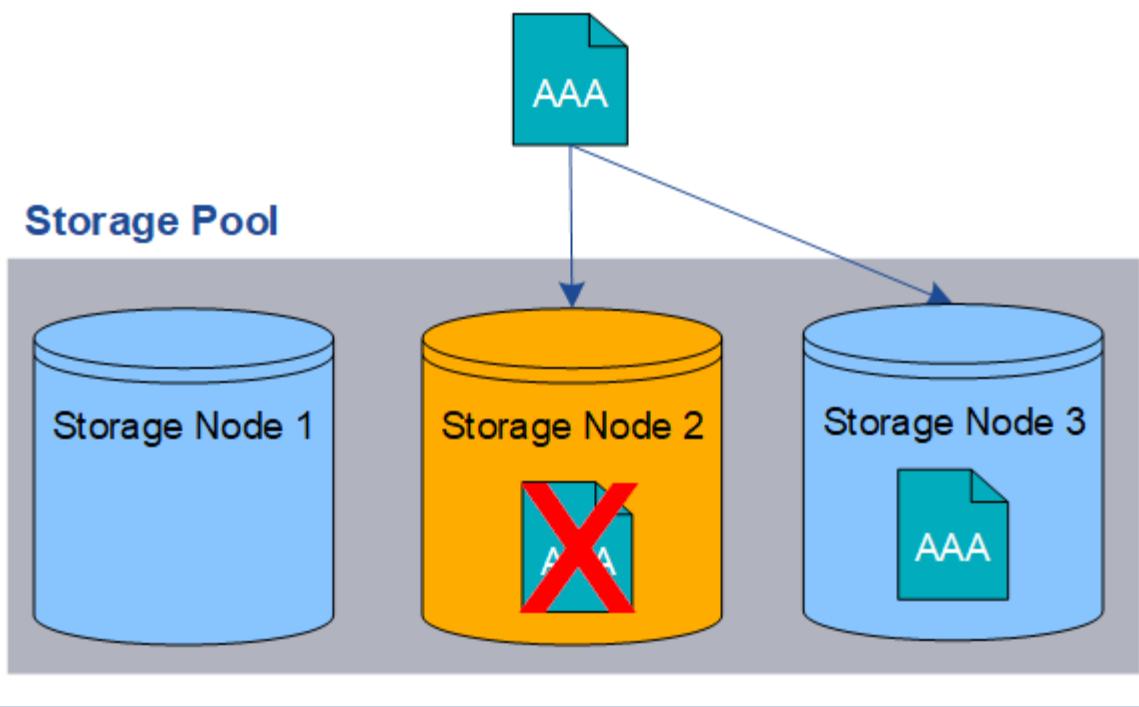
Cuando una regla de ILM crea solo una copia replicada de un objeto, se vuelve inaccesible cuando el nodo de almacenamiento no está disponible. En este ejemplo, perderá temporalmente el acceso al objeto AAA siempre que el nodo de almacenamiento 2 esté desconectado, como durante una actualización u otro procedimiento de mantenimiento. Perderá el objeto AAA completamente si falla el nodo de almacenamiento 2.

## Make 1 Copy



Para evitar la pérdida de datos de objetos, siempre debe realizar al menos dos copias de todos los objetos que deseé proteger con replicación. Si existen dos o más copias, puede seguir teniendo acceso al objeto si un nodo de almacenamiento falla o se desconecta.

## Make 2 Copies



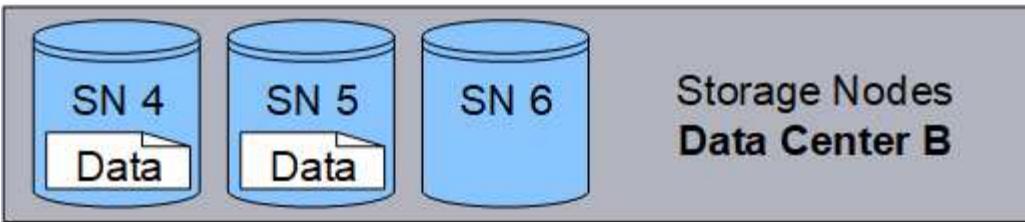
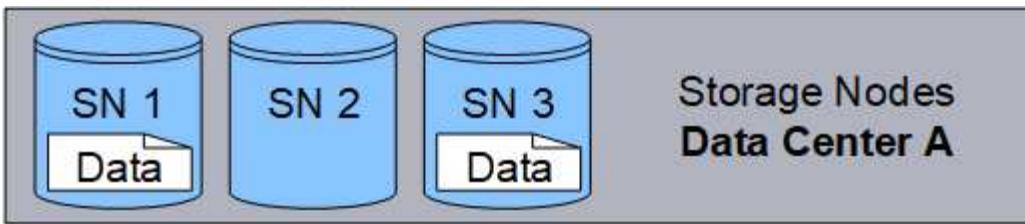
### ¿Qué es el código de borrado?

El código de borrado es uno de los dos métodos que utiliza StorageGRID para almacenar datos de objetos. Cuando los objetos coinciden con una regla de ILM que utiliza código de borrado, esos objetos se dividen en fragmentos de datos, se calculan fragmentos de paridad adicionales y cada fragmento se almacena en un nodo de almacenamiento diferente.

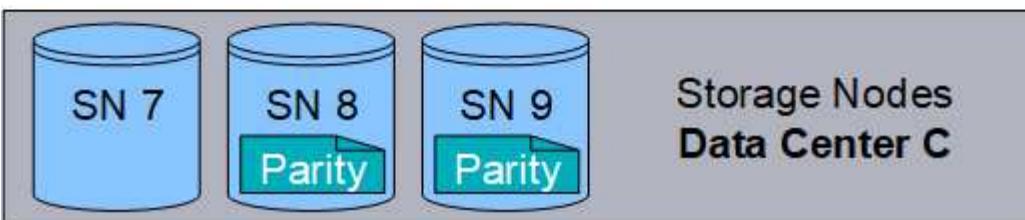
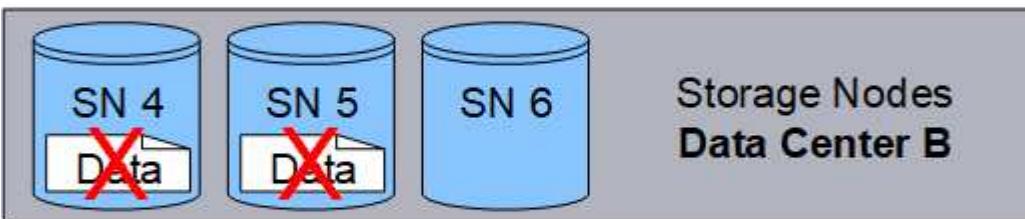
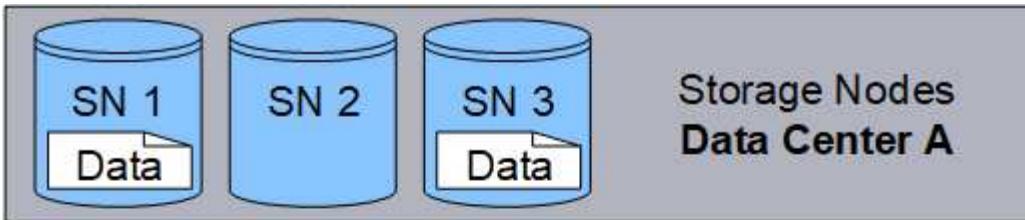
Cuando se accede a un objeto, se vuelve a ensamblar utilizando los fragmentos almacenados. Si un dato o un fragmento de paridad se corrompen o se pierden, el algoritmo de código de borrado puede recrear ese fragmento con un subconjunto de los datos restantes y los fragmentos de paridad.

Al crear reglas de ILM, StorageGRID crea perfiles de código de borrado que admiten esas reglas. Puede ver una lista de perfiles de codificación de borrado, "[cambie el nombre de un perfil de código de borrado](#)", o "[Desactive un perfil de código de borrado si actualmente no se utiliza en ninguna regla de ILM](#)".

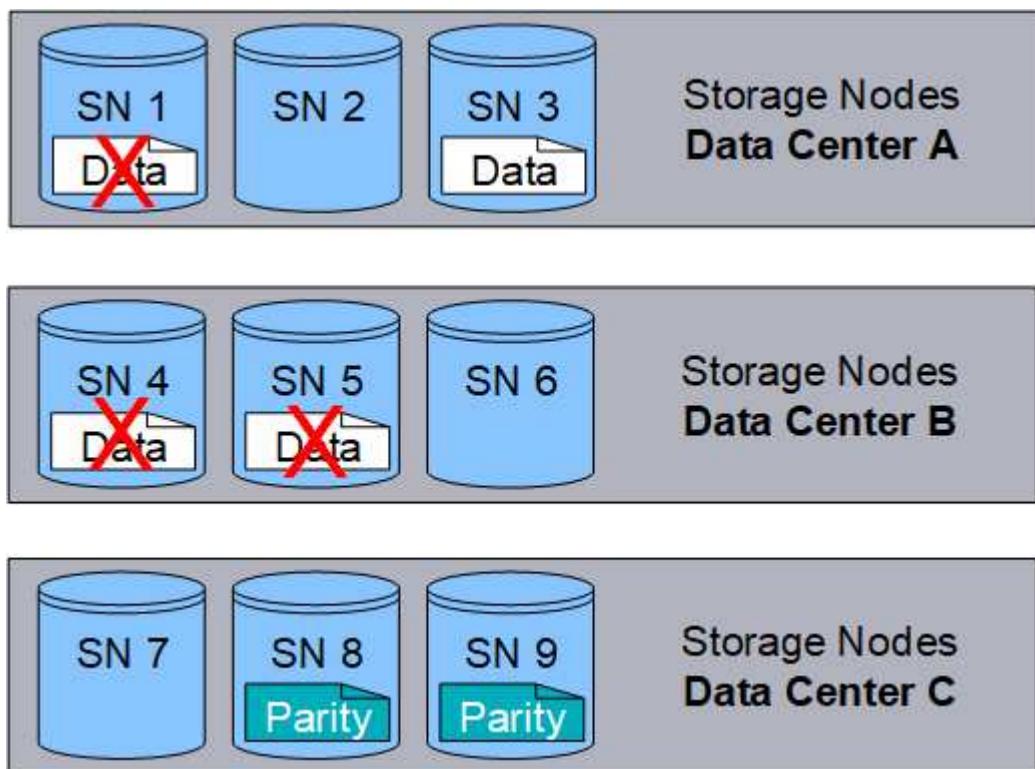
En el siguiente ejemplo, se muestra el uso de un algoritmo de codificación de borrado en los datos de un objeto. En este ejemplo, la regla ILM utiliza un esquema de codificación de borrado 4+2. Cada objeto se divide en cuatro fragmentos de datos iguales y dos fragmentos de paridad se calculan a partir de los datos del objeto. Cada uno de los seis fragmentos se almacena en un nodo diferente en tres sitios de centro de datos para proporcionar protección de datos ante fallos de nodos o pérdidas de sitios.



El esquema de codificación de borrado 4+2 se puede configurar de varias maneras. Por ejemplo, puede configurar un pool de almacenamiento de un único sitio que contenga seis nodos de almacenamiento. Para "protección contra pérdida de sitios", Puede utilizar un pool de almacenamiento que contenga tres sitios con tres nodos de almacenamiento en cada sitio. Un objeto se puede recuperar siempre que cuatro de los seis fragmentos (datos o paridad) permanezcan disponibles. Se pueden perder hasta dos fragmentos sin perder los datos del objeto. Si se pierde un sitio entero, el objeto aún puede recuperarse o repararse, siempre que se pueda acceder a todos los demás fragmentos.



Si se pierden más de dos nodos de almacenamiento, el objeto no se puede recuperar.



#### Información relacionada

- "[¿Qué es la replicación?](#)"
- "[¿Qué es un pool de almacenamiento?](#)"
- "[¿Qué son los esquemas de código de borrado?](#)"
- "[Cambio el nombre de un perfil de código de borrado](#)"
- "[Desactivar un perfil de código de borrado](#)"

### ¿Qué son los esquemas de código de borrado?

Los esquemas de codificación de borrado controlan cuántos fragmentos de datos se crean y cuántos fragmentos de paridad se crean para cada objeto.

Cuando configura el perfil de código de borrado para una regla de ILM, debe seleccionar un esquema de código de borrado disponible en función del número de nodos de almacenamiento y sitios que componen el pool de almacenamiento que piensa usar.

El sistema StorageGRID utiliza el algoritmo de codificación de borrado Reed-Solomon. El algoritmo divide un objeto en  $k$  fragmentos de datos y cálculos  $m$  fragmentos de paridad. La  $k + m = n$  los fragmentos se distribuyen  $n$  Nodos de almacenamiento para proporcionar protección de datos. Un objeto puede soportar hasta  $m$  fragmentos perdidos o corruptos. Para recuperar o reparar un objeto,  $k$  se necesitan fragmentos.

Cuando seleccione el pool de almacenamiento que se utilizará en una regla que creará una copia con código de borrado, utilice las siguientes directrices para los pools de almacenamiento:

- El pool de almacenamiento debe incluir tres o más sitios, o exactamente un sitio.



No se puede usar código de borrado si el pool de almacenamiento incluye dos sitios.

- [Esquemas de codificación de borrado para pools de almacenamiento que contengan tres o más sitios](#)
- [Esquemas de codificación de borrado para pools de almacenamiento in situ](#)
- No utilice un pool de almacenamiento que incluya el sitio predeterminado, Todos los sitios.
- El pool de almacenamiento debe incluir al menos  $k+m+1$  Nodos de almacenamiento.

El número mínimo necesario de nodos de almacenamiento es  $k+m$ . Sin embargo, tener al menos un nodo de almacenamiento adicional puede ayudar a evitar fallos de ingestión o errores de gestión de la vida útil si un nodo de almacenamiento necesario no está disponible temporalmente.

La sobrecarga de almacenamiento de un esquema de código de borrado se calcula dividiendo el número de fragmentos de paridad ( $m$ ) por el número de fragmentos de datos ( $k$ ). Puede utilizar la sobrecarga del almacenamiento para calcular cuánto espacio en disco necesita cada objeto con código de borrado:

$$\text{disk space} = \text{object size} + (\text{object size} * \text{storage overhead})$$

Por ejemplo, si almacena un objeto de 10 MB mediante el esquema 4+2 (que tiene un 50% de sobrecarga de almacenamiento), el objeto consume 15 MB de almacenamiento de cuadrícula. Si almacena el mismo objeto de 10 MB con el esquema 6+2 (que tiene un 33% de sobrecarga de almacenamiento), el objeto consume aproximadamente 13.3 MB.

Seleccione el esquema de código de borrado que tenga el valor total más bajo de  $k+m$  que satisface sus necesidades. Los esquemas de código de borrado con un número menor de fragmentos por lo general son más eficientes computacionalmente, ya que se crean y distribuyen menos fragmentos (o se recuperan) por objeto, pueden ofrecer un mejor rendimiento debido al tamaño de fragmento más grande y pueden requerir que se añadan menos nodos en una expansión cuando se necesita más almacenamiento. (Para obtener información sobre cómo planificar una expansión de almacenamiento, consulte ["Instrucciones para ampliar StorageGRID".](#))

### **Esquemas de codificación de borrado para pools de almacenamiento que contengan tres o más sitios**

En la siguiente tabla se describen los esquemas de codificación de borrado que admite actualmente StorageGRID para pools de almacenamiento que incluyen tres o más sitios. Todos estos esquemas proporcionan protección contra pérdida de sitio. Se puede perder un sitio y el objeto seguirá siendo accesible.

Para esquemas de código de borrado que ofrecen protección contra pérdida de sitio, la cantidad recomendada de nodos de almacenamiento en el pool de almacenamiento supera  $k+m+1$ . Dado que cada sitio requiere un mínimo de tres nodos de almacenamiento.

Esquema de codificación de borrado ( $k+m$ )	Número mínimo de sitios implementados	Número recomendado de nodos de almacenamiento en cada sitio	Número total recomendado de nodos de almacenamiento	¿Protección contra pérdida de sitio?	Gastos generales de almacenamiento
4+2	3	3	9	Sí	50 %
6+2	4	3	12	Sí	33 %

Esquema de codificación de borrado ( $k+m$ )	Número mínimo de sitios implementados	Número recomendado de nodos de almacenamiento en cada sitio	Número total recomendado de nodos de almacenamiento	¿Protección contra pérdida de sitio?	Gastos generales de almacenamiento
8+2	5	3	15	Sí	25 %
6+3	3	4	12	Sí	50 %
9+3	4	4	16	Sí	33 %
2+1	3	3	9	Sí	50 %
4+1	5	3	15	Sí	25 %
6+1	7	3	21	Sí	17 %
7+5	3	5	15	Sí	71 %



StorageGRID requiere un mínimo de tres nodos de almacenamiento por sitio. Para utilizar el esquema 7+5, cada sitio requiere un mínimo de cuatro nodos de almacenamiento. Se recomienda usar cinco nodos de almacenamiento por sitio.

Al seleccionar un esquema de codificación de borrado que proporcione protección al sitio, equilibre la importancia relativa de los siguientes factores:

- **Número de fragmentos:** El rendimiento y la flexibilidad de expansión son generalmente mejores cuando el número total de fragmentos es menor.
- **Tolerancia a fallos:** La tolerancia a fallos se incrementa al tener más segmentos de paridad (es decir, cuando  $m$  tiene un valor más alto.)
- **Tráfico de red:** Al recuperarse de fallos, utilizando un esquema con más fragmentos (es decir, un total más alto para  $k+m$ ) crea más tráfico de red.
- **Gastos generales de almacenamiento:** Los esquemas con mayor sobrecarga requieren más espacio de almacenamiento por objeto.

Por ejemplo, al decidir entre un esquema 4+2 y un esquema 6+3 (que ambos tienen un 50% de gastos generales de almacenamiento), seleccione el esquema 6+3 si se requiere tolerancia a fallos adicional. Seleccione el esquema 4+2 si los recursos de red están limitados. Si todos los demás factores son iguales, seleccione 4+2 porque tiene un número total menor de fragmentos.



Si no está seguro de qué esquema usar, seleccione 4+2 o 6+3, o póngase en contacto con el servicio de asistencia técnica.

## Esquemas de codificación de borrado para pools de almacenamiento in situ

Un pool de almacenamiento in situ admite todos los esquemas de codificación de borrado definidos para tres o más sitios, siempre y cuando el sitio tenga suficientes nodos de almacenamiento.

El número mínimo necesario de nodos de almacenamiento es  $k+m$ , pero una piscina de almacenamiento con  $k+m+1$  se recomienda los nodos de almacenamiento. Por ejemplo, el esquema de codificación de borrado  $2+1$  requiere un pool de almacenamiento con un mínimo de tres nodos de almacenamiento, pero se recomienda cuatro nodos de almacenamiento.

Esquema de codificación de borrado ( $k+m$ )	Número mínimo de nodos de almacenamiento	Número recomendado de nodos de almacenamiento	Gastos generales de almacenamiento
4+2	6	7	50 %
6+2	8	9	33 %
8+2	10	11	25 %
6+3	9	10	50 %
9+3	12	13	33 %
2+1	3	4	50 %
4+1	5	6	25 %
6+1	7	8	17 %
7+5	12	13	71 %

## Ventajas, desventajas y requisitos de codificación de borrado

Antes de decidir si se debe utilizar la replicación o el código de borrado para proteger los datos de objetos frente a pérdidas, debe comprender las ventajas, las desventajas y los requisitos para la codificación de borrado.

### Ventajas de la codificación de borrado

En comparación con la replicación, la codificación de borrado ofrece una mayor fiabilidad, disponibilidad y eficiencia del almacenamiento.

- **Confiabilidad:** La fiabilidad se mide en términos de tolerancia a fallos, es decir, el número de fallos simultáneos que se pueden sostener sin pérdida de datos. Con la replicación, se almacenan varias copias idénticas en diferentes nodos y entre sitios. Con el código de borrado, un objeto se codifica en fragmentos de datos y de paridad, y se distribuye entre muchos nodos y sitios. Esta dispersión proporciona protección frente a fallos del sitio y del nodo. En comparación con la replicación, la codificación de borrado proporciona una mayor fiabilidad con costes de almacenamiento comparables.
- **Disponibilidad:** La disponibilidad se puede definir como la capacidad de recuperar objetos si los nodos de almacenamiento fallan o se vuelven inaccesibles. En comparación con la replicación, la codificación de borrado proporciona una mayor disponibilidad con costes de almacenamiento comparables.
- **Eficiencia del almacenamiento:** Para niveles similares de disponibilidad y fiabilidad, los objetos protegidos mediante codificación de borrado consumen menos espacio en disco que los mismos objetos si

están protegidos mediante replicación. Por ejemplo, un objeto de 10 MB que se replica en dos sitios consume 20 MB de espacio en disco (dos copias), mientras que un objeto que se elimina en tres sitios con un esquema de codificación de borrado 6+3 solo consume 15 MB de espacio en disco.



El espacio en disco para los objetos codificados de borrado se calcula como el tamaño del objeto más la sobrecarga del almacenamiento. El porcentaje de sobrecarga del almacenamiento es el número de fragmentos de paridad dividido por el número de fragmentos de datos.

## Desventajas del código de borrado

En comparación con la replicación, los códigos de borrado tienen las siguientes desventajas:

- Se recomienda un mayor número de nodos de almacenamiento y sitios, en función del esquema de código de borrado. Por el contrario, si replica datos de objetos, solo necesita un nodo de almacenamiento para cada copia. Consulte "[Esquemas de código de borrado para pools de almacenamiento que contienen tres o más sitios](#)" y.. "[Esquemas de código de borrado para pools de almacenamiento de un sitio](#)".
- Aumento del coste y de la complejidad de las ampliaciones del almacenamiento. Para expandir una implementación que utiliza replicación, debe agregar capacidad de almacenamiento en cada ubicación donde se realicen copias de objetos. Para ampliar una puesta en marcha que utilice código de borrado, debe tener en cuenta el esquema de codificación de borrado y el grado de llenado de los nodos de almacenamiento existentes. Por ejemplo, si espera a que los nodos existentes estén llenos al 100 %, debe agregar al menos  $k+m$  Nodos de almacenamiento, pero si amplía cuando los nodos existentes estén llenos al 70 %, puede añadir dos nodos por sitio y seguir maximizando la capacidad de almacenamiento útil. Para obtener más información, consulte "[Añada capacidad de almacenamiento para objetos codificados de borrado](#)".
- Al utilizar códigos de borrado en ubicaciones distribuidas geográficamente, aumenta la latencia de recuperación. Los fragmentos de objeto para un objeto que se codifica con borrado y se distribuyen en sitios remotos tardan más en recuperarse a través de conexiones WAN que los objetos que se replican y están disponibles localmente (el mismo sitio al que se conecta el cliente).
- Al utilizar la codificación de borrado en ubicaciones distribuidas geográficamente, se está utilizando más el tráfico de red WAN para restauraciones y reparaciones, especialmente en objetos que se recuperan con frecuencia o para reparaciones de objetos a través de conexiones de red WAN.
- Cuando se utiliza la codificación de borrado en varios sitios, el rendimiento máximo del objeto se reduce drásticamente a medida que aumenta la latencia de red entre sitios. Esta disminución se debe a la correspondiente disminución del rendimiento de la red TCP, que afecta a la rapidez con la que el sistema StorageGRID puede almacenar y recuperar fragmentos de objeto.
- Mayor uso de recursos de computación.

## Cuándo se debe utilizar la codificación de borrado

El código de borrado se ajusta mejor a los siguientes requisitos:

- Los objetos tienen un tamaño superior a 1 MB.



El código de borrado se adapta mejor a los objetos de más de 1 MB. No use el código de borrado para objetos de menos de 200 KB para evitar la sobrecarga de gestionar fragmentos de código de borrado muy pequeños.

- Almacenamiento a largo plazo o en frío para contenido que se recupera con poca frecuencia.

- Alta disponibilidad y fiabilidad de los datos.
- Protección frente a fallos completos de sitios y nodos.
- Eficiencia del almacenamiento.
- Puestas en marcha de un único sitio que requieren protección de datos eficiente con solo una copia codificada por borrado en lugar de múltiples copias replicadas.
- Puestas en marcha de varios sitios en las que la latencia entre sitios es inferior a 100 ms.

## Cómo se determina la retención de objetos

StorageGRID ofrece opciones tanto para los administradores de grid como para los usuarios individuales de inquilino para especificar el tiempo que se tarda en almacenar los objetos. En general, cualquier instrucción de retención proporcionada por un usuario inquilino tiene prioridad sobre las instrucciones de retención proporcionadas por el administrador de grid.

### Cómo los usuarios de inquilinos controlan la retención de objetos

Los usuarios de inquilinos tienen tres formas principales de controlar cuánto tiempo se almacenan los objetos en StorageGRID:

- Si la configuración global de Object Lock está habilitada para el grid, los usuarios inquilinos S3 pueden crear bloques con S3 Object Lock habilitado y, a continuación, utilizar la API REST de S3 para especificar la configuración de retención hasta la fecha y la conservación legal de cada versión de objeto añadida a ese bloque.
  - Una versión de objeto que está bajo una conservación legal no se puede eliminar con ningún método.
  - Antes de que se alcance la fecha de retención hasta la versión de un objeto, esa versión no se puede eliminar con ningún método.
  - Los objetos en bloques con S3 Object Lock habilitado son mantenidos por ILM "eternamente". Sin embargo, una vez alcanzada la fecha de retención hasta la fecha, una solicitud de cliente puede eliminar una versión de objeto o la expiración del ciclo de vida de la cuchara. Consulte "[Gestione objetos con S3 Object Lock](#)".
- Los usuarios de inquilinos S3 pueden añadir una configuración del ciclo de vida a sus bloques que especifica una acción de caducidad. Si existe un ciclo de vida de un bloque, StorageGRID almacena un objeto hasta que se cumpla la fecha o el número de días especificados en la acción Expiración, a menos que el cliente elimine primero el objeto. Consulte "[Cree una configuración del ciclo de vida de S3](#)".
- Un cliente S3 o Swift puede emitir una solicitud de eliminación de objeto. StorageGRID siempre prioriza las solicitudes de eliminación de clientes por encima del ciclo de vida de los bloques S3 o ILM al determinar si se debe eliminar o conservar un objeto.

### Cómo los administradores de grid controlan la retención de objetos

Los administradores de grid utilizan las instrucciones de colocación de ILM para controlar la duración de los objetos almacenados. Cuando una regla de ILM coincide con los objetos, StorageGRID almacena esos objetos hasta que haya transcurrido el último periodo de tiempo de la regla de ILM. Los objetos se conservan indefinidamente si se especifica "eternamente" para las instrucciones de colocación.

Independientemente de quién controle cuánto tiempo se retienen los objetos, la configuración de ILM controla qué tipos de copias de objetos (replicadas o codificadas de borrado) se almacenan y dónde se encuentran las

copias (nodos de almacenamiento, pools de almacenamiento en cloud o nodos de archivado).

## Cómo interaccionan el ciclo de vida de bloque y ILM de S3

La acción de caducidad en un ciclo de vida de bloque de S3 siempre anula la configuración de ILM. Como resultado, es posible que un objeto se conserve en la cuadrícula aunque hayan caducado las instrucciones de gestión del ciclo de vida de la información relativas a la ubicación del objeto.

### Ejemplos para la retención de objetos

Para comprender mejor las interacciones entre S3 Object Lock, la configuración del ciclo de vida de bloques, las solicitudes de eliminación de clientes y ILM, tenga en cuenta los siguientes ejemplos.

#### Ejemplo 1: El ciclo de vida de un bloque de S3 mantiene los objetos durante más tiempo que ILM

##### ILM

Almacene dos copias por 1 año (365 días)

##### Ciclo de vida del cucharón

Caducidad de objetos en 2 años (730 días)

##### Resultado

StorageGRID almacena el objeto durante 730 días. StorageGRID utiliza la configuración del ciclo de vida de los bloques para determinar si se debe eliminar o conservar un objeto.



Si el ciclo de vida de un bloque especifica que los objetos se deben conservar durante más tiempo del ciclo de vida de la información especificado por ILM, StorageGRID sigue usando las instrucciones de colocación de ILM al determinar el número y el tipo de copias que se deben almacenar. En este ejemplo, se seguirán almacenando dos copias del objeto en StorageGRID de los días 366 a 730.

#### Ejemplo 2: El ciclo de vida de bloque de S3 caduca los objetos antes de ILM

##### ILM

Almacene dos copias durante 2 años (730 días)

##### Ciclo de vida del cucharón

Caducar objetos en un año (365 días)

##### Resultado

StorageGRID elimina ambas copias del objeto después del día 365.

#### Ejemplo 3: La eliminación de clientes anula el ciclo de vida del bloque y el ILM

##### ILM

Almacenar dos copias en nodos de almacenamiento «para siempre»

##### Ciclo de vida del cucharón

Caducidad de objetos en 2 años (730 días)

## **Solicitud de eliminación de cliente**

Emitido el día 400

### **Resultado**

StorageGRID elimina ambas copias del objeto el día 400 en respuesta a la solicitud de eliminación del cliente.

## **Ejemplo 4: El bloqueo de objetos S3 anula la solicitud de eliminación del cliente**

### **Bloqueo de objetos de S3**

La fecha de retención hasta la versión de un objeto es 2026-03-31. No existe un derecho legal.

### **Regla de ILM que cumpla con las normativas**

Almacenar dos copias en nodos de almacenamiento «para siempre».

### **Solicitud de eliminación de cliente**

Emitido el 2024-03-31.

### **Resultado**

StorageGRID no eliminará la versión del objeto porque la fecha de retención hasta todavía está a 2 años.

## **Cómo se eliminan los objetos**

StorageGRID puede eliminar objetos en respuesta directa a una solicitud del cliente o de forma automática como resultado del vencimiento del ciclo de vida de un bloque de S3 o de los requisitos de la política de ILM. Comprender las diferentes formas en que se pueden eliminar los objetos y el modo en que StorageGRID gestiona las solicitudes de eliminación puede ayudarle a gestionar los objetos de forma más eficaz.

StorageGRID puede utilizar uno de estos dos métodos para eliminar objetos:

- Eliminación síncrona: Cuando StorageGRID recibe una solicitud de eliminación de cliente, todas las copias de los objetos se eliminan de inmediato. Se informa al cliente de que la eliminación se ha realizado correctamente una vez eliminadas las copias.
- Los objetos se ponen en cola para eliminación: Cuando StorageGRID recibe una solicitud de eliminación, el objeto se pone en cola para su eliminación y se informa al cliente inmediatamente de que esta se ha eliminado correctamente. Las copias de objetos se eliminan más adelante mediante el procesamiento de ILM en segundo plano.

Cuando se eliminan objetos, StorageGRID utiliza el método que optimiza el rendimiento de eliminación, minimiza las posibles acumulaciones de eliminación y libera espacio que se libera con mayor rapidez.

La tabla resume cuándo StorageGRID utiliza cada método.

Método de eliminación	Cuando se utilice
Los objetos se mantienen en la cola para su eliminación	<p>Cuando <b>cualquiera</b> de las siguientes condiciones se cumple:</p> <ul style="list-style-type: none"> <li>La eliminación automática de objetos ha sido activada por uno de los siguientes eventos: <ul style="list-style-type: none"> <li>Se ha alcanzado la fecha de caducidad o el número de días en la configuración del ciclo de vida de un bloque de S3.</li> <li>El último periodo de tiempo especificado en una regla de ILM transcurre.</li> </ul> </li> <li><b>Nota:</b> Los objetos en un depósito que tiene S3 Object Lock habilitado no se pueden eliminar si están bajo una retención legal o si se ha especificado una fecha de retención hasta la fecha pero aún no se cumplen.</li> <li>Un cliente de S3 o Swift solicita la eliminación y se debe cumplir una o varias de estas condiciones: <ul style="list-style-type: none"> <li>Las copias no se pueden eliminar en 30 segundos porque, por ejemplo, una ubicación de objetos no está disponible en este momento.</li> <li>Las colas de eliminación en segundo plano están inactivas.</li> </ul> </li> </ul>
Los objetos se quitan de inmediato (eliminación síncrona)	<p>Cuando un cliente S3 o Swift realiza una solicitud de eliminación y <b>se cumplen todas</b> las siguientes condiciones:</p> <ul style="list-style-type: none"> <li>Todas las copias se pueden eliminar en 30 segundos.</li> <li>Las colas de eliminación en segundo plano contienen objetos que se van a procesar.</li> </ul>

Cuando los clientes S3 o Swift realizan solicitudes de eliminación, StorageGRID empieza añadiendo objetos a la cola de eliminación. A continuación, cambia a realizar una eliminación síncrona. Asegurarse de que la cola de eliminación en segundo plano tiene objetos que procesar permite a StorageGRID procesar las eliminaciones de forma más eficaz, especialmente en los clientes de baja concurrencia, mientras que ayuda a evitar que los clientes eliminen las copias de seguridad.

## Tiempo necesario para eliminar objetos

La forma en que StorageGRID elimina los objetos puede afectar a la forma en la que aparece el sistema:

- Cuando StorageGRID realiza la eliminación síncrona, StorageGRID puede tardar hasta 30 segundos en devolver un resultado al cliente. Esto significa que la eliminación puede parecer más lenta, aunque en realidad se eliminan copias más rápidamente de lo que están cuando StorageGRID pone en cola objetos para su eliminación.
- Si supervisa de cerca el rendimiento de eliminación durante una eliminación masiva, puede observar que la tasa de eliminación aparece como lenta después de eliminar un cierto número de objetos. Este cambio ocurre cuando StorageGRID pasa de poner objetos en cola para su eliminación a realizar una eliminación síncrona. La reducción aparente en la tasa de eliminación no significa que las copias de objetos se van a eliminar más lentamente. Por el contrario, indica que, en promedio, ahora se libera espacio con más rapidez.

Si elimina un gran número de objetos y la prioridad es liberar espacio rápidamente, considere la posibilidad de usar una solicitud de cliente para eliminar objetos en lugar de eliminarlos con ILM u otros métodos. En general, el espacio se libera más rápidamente cuando los clientes lo eliminan, ya que StorageGRID puede utilizar la eliminación síncrona.

La cantidad de tiempo necesario para liberar espacio después de eliminar un objeto depende de varios factores:

- Si las copias de objetos se eliminan de forma síncrona o se ponen en cola para su eliminación más adelante (para solicitudes de eliminación de clientes).
- Otros factores, como el número de objetos de la cuadrícula o la disponibilidad de los recursos de grid cuando las copias de objetos se colocan en cola para su eliminación (tanto para las eliminaciones del cliente como para otros métodos).

## Cómo se eliminan los objetos con versiones de S3

Cuando se habilita el control de versiones para un bloque de S3, StorageGRID sigue el comportamiento de Amazon S3 al responder a las solicitudes de eliminación, ya provenga de un cliente S3, el vencimiento de un ciclo de vida de un bloque de S3 o los requisitos de la política de ILM.

Cuando los objetos se versionan, las solicitudes de eliminación de objetos no eliminan la versión actual del objeto y no liberan espacio. En su lugar, una solicitud de eliminación de objetos crea un marcador de eliminación como la versión actual del objeto, lo que hace que la versión anterior del objeto sea “no actual”.

Aunque el objeto no se haya quitado, StorageGRID se comporta como si la versión actual del objeto ya no estuviera disponible. Las solicitudes a ese objeto devuelven 404 NotFound. Sin embargo, debido a que los datos de objeto no actuales no se han eliminado, las solicitudes que especifican una versión no actual del objeto pueden tener éxito.

Para liberar espacio al suprimir objetos con versiones, utilice una de las siguientes opciones:

- **Solicitud de cliente S3:** Especifique el ID de versión de objeto en la solicitud de ELIMINACIÓN DE objeto S3 (`DELETE /object?versionId=ID`). Tenga en cuenta que esta solicitud sólo elimina copias de objetos para la versión especificada (las otras versiones todavía ocupan espacio).
- **Ciclo de vida del cucharón:** Utilice `NoncurrentVersionExpiration` acción en la configuración del ciclo de vida del bloque. Cuando se cumple el número de días sin `currentDays` especificado, StorageGRID elimina permanentemente todas las copias de las versiones de objetos no actuales. Estas versiones de objetos no se pueden recuperar.

La `NewerNoncurrentVersions` La acción en la configuración del ciclo de vida del bloque especifica el número de versiones no actuales retenidas en un segmento S3 con versiones. Si hay más versiones no corrientes que `NewerNoncurrentVersions` Especifica, StorageGRID elimina las versiones anteriores una vez transcurrido el valor `NoncurrentDays`. La `NewerNoncurrentVersions` El umbral anula las reglas de ciclo de vida proporcionadas por ILM, lo que significa que un objeto no actual con una versión dentro del `NewerNoncurrentVersions` El umbral se conserva si ILM solicita su eliminación.

- **ILM: "Clone la política activa"** Y añada dos reglas de ILM a la nueva política propuesta:
  - Primera regla: Use “Tiempo no corriente” como el Tiempo de referencia para coincidir con las versiones no actuales del objeto. Pulg ["Paso 1 \(introduzca detalles\) del asistente Create an ILM Rule"](#), Seleccione **Sí** para la pregunta, “Aplica esta regla solo a versiones de objetos más antiguas (en cubos S3 con control de versiones activado)?”
  - Segunda regla: Utilice **tiempo de ingestión** para que coincida con la versión actual. La regla “Tiempo no

corriente” debe aparecer en la política sobre la regla **Tiempo de ingestión**.

### Cómo se eliminan los marcadores de borrado de S3

Cuando se elimina un objeto versionado, StorageGRID crea un marcador de eliminación como la versión actual del objeto. Para eliminar el marcador de eliminación de cero bytes del depósito, el cliente S3 debe eliminar explícitamente la versión del objeto. ILM, las reglas de ciclo de vida de los bloques no se quitan los marcadores de eliminación ni los objetos de las operaciones de bloque.

#### Información relacionada

- ["USAR LA API DE REST DE S3"](#)
- ["Ejemplo 4: Reglas de ILM y políticas para objetos con versiones de S3"](#)

## **Información de copyright**

Copyright © 2025 NetApp, Inc. Todos los derechos reservados. Imprimido en EE. UU. No se puede reproducir este documento protegido por copyright ni parte del mismo de ninguna forma ni por ningún medio (gráfico, electrónico o mecánico, incluidas fotocopias, grabaciones o almacenamiento en un sistema de recuperación electrónico) sin la autorización previa y por escrito del propietario del copyright.

El software derivado del material de NetApp con copyright está sujeto a la siguiente licencia y exención de responsabilidad:

ESTE SOFTWARE LO PROPORCIONA NETAPP «TAL CUAL» Y SIN NINGUNA GARANTÍA EXPRESA O IMPLÍCITA, INCLUYENDO, SIN LIMITAR, LAS GARANTÍAS IMPLÍCITAS DE COMERCIALIZACIÓN O IDONEIDAD PARA UN FIN CONCRETO, CUYA RESPONSABILIDAD QUEDA EXIMIDA POR EL PRESENTE DOCUMENTO. EN NINGÚN CASO NETAPP SERÁ RESPONSABLE DE NINGÚN DAÑO DIRECTO, INDIRECTO, ESPECIAL, EJEMPLAR O RESULTANTE (INCLUYENDO, ENTRE OTROS, LA OBTENCIÓN DE BIENES O SERVICIOS SUSTITUTIVOS, PÉRDIDA DE USO, DE DATOS O DE BENEFICIOS, O INTERRUPCIÓN DE LA ACTIVIDAD EMPRESARIAL) CUALQUIERA SEA EL MODO EN EL QUE SE PRODUJERON Y LA TEORÍA DE RESPONSABILIDAD QUE SE APLIQUE, YA SEA EN CONTRATO, RESPONSABILIDAD OBJETIVA O AGRAVIO (INCLUIDA LA NEGLIGENCIA U OTRO TIPO), QUE SURJAN DE ALGÚN MODO DEL USO DE ESTE SOFTWARE, INCLUSO SI HUBIEREN SIDO ADVERTIDOS DE LA POSIBILIDAD DE TALES DAÑOS.

NetApp se reserva el derecho de modificar cualquiera de los productos aquí descritos en cualquier momento y sin aviso previo. NetApp no asume ningún tipo de responsabilidad que surja del uso de los productos aquí descritos, excepto aquello expresamente acordado por escrito por parte de NetApp. El uso o adquisición de este producto no lleva implícita ninguna licencia con derechos de patente, de marcas comerciales o cualquier otro derecho de propiedad intelectual de NetApp.

Es posible que el producto que se describe en este manual esté protegido por una o más patentes de EE. UU., patentes extranjeras o solicitudes pendientes.

**LEYENDA DE DERECHOS LIMITADOS:** el uso, la copia o la divulgación por parte del gobierno están sujetos a las restricciones establecidas en el subpárrafo (b)(3) de los derechos de datos técnicos y productos no comerciales de DFARS 252.227-7013 (FEB de 2014) y FAR 52.227-19 (DIC de 2007).

Los datos aquí contenidos pertenecen a un producto comercial o servicio comercial (como se define en FAR 2.101) y son propiedad de NetApp, Inc. Todos los datos técnicos y el software informático de NetApp que se proporcionan en este Acuerdo tienen una naturaleza comercial y se han desarrollado exclusivamente con fondos privados. El Gobierno de EE. UU. tiene una licencia limitada, irrevocable, no exclusiva, no transferible, no sublicenciable y de alcance mundial para utilizar los Datos en relación con el contrato del Gobierno de los Estados Unidos bajo el cual se proporcionaron los Datos. Excepto que aquí se disponga lo contrario, los Datos no se pueden utilizar, desvelar, reproducir, modificar, interpretar o mostrar sin la previa aprobación por escrito de NetApp, Inc. Los derechos de licencia del Gobierno de los Estados Unidos de América y su Departamento de Defensa se limitan a los derechos identificados en la cláusula 252.227-7015(b) de la sección DFARS (FEB de 2014).

## **Información de la marca comercial**

NETAPP, el logotipo de NETAPP y las marcas que constan en <http://www.netapp.com/TM> son marcas comerciales de NetApp, Inc. El resto de nombres de empresa y de producto pueden ser marcas comerciales de sus respectivos propietarios.