

Desperdicio Invisible en AWS S3

El impacto oculto de obviar costos de almacenamiento en S3.



Presenta:

Danny Banol - EPAM Systems Engineer

Octubre 2025



COMUNIDAD

Cloud Crafters

En CloudCrafters, los entusiastas de la tecnología se unen para explorar el vasto mundo de la computación en la nube. Únete a nosotros para compartir conocimiento, fomentar la innovación y conectar con personas afines de todo el mundo. Juntos, impulsemos el futuro de la tecnología.



Que veremos...

Identificar cual es ese desperdicio invisible, desde
multiparts incompletos, versiones viejas y data duplicada --
Contrastes entre diferentes escenarios para ilustrar
diferentes patrones de acceso y el impacto en los costos --
Storage classes, lifecycles, intelligent tiering entre otras
herramientas -- Casos de la vida real en FinOps y ahorro

- Por que deberia importarnos
- Fuentes de costos ocultos
- Casos de uso y patrones
- Estrategias de optimizacion
- Ejemplos y lo aprendido
- Mejores practicas en FinOps





INTRODUCCION

Por que deberia importarnos ese desperdicio invisible en S3?

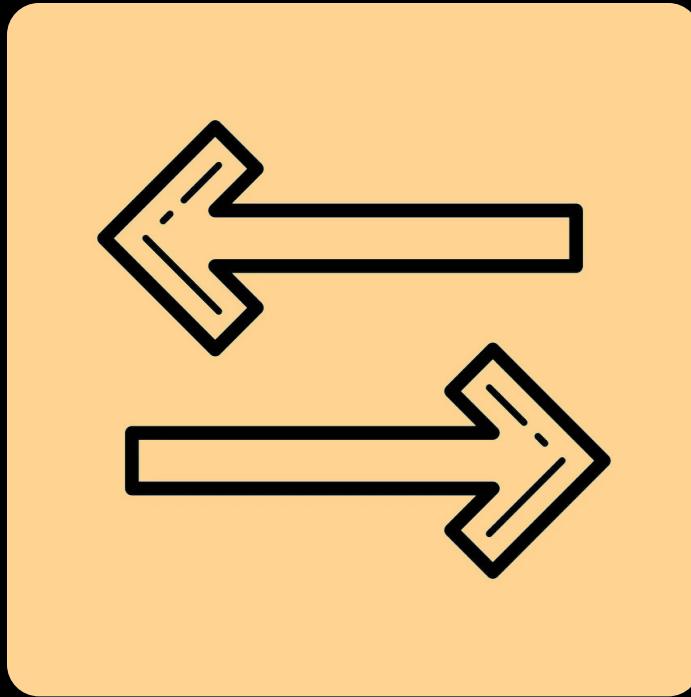
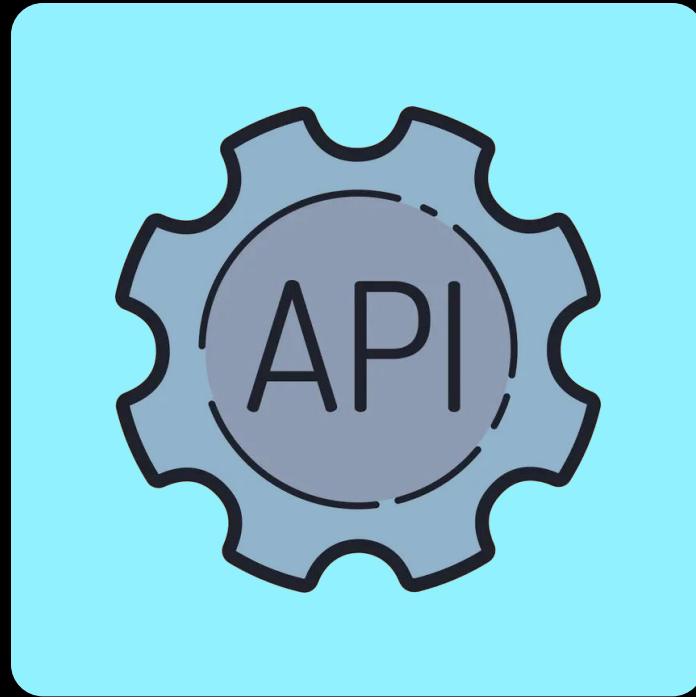
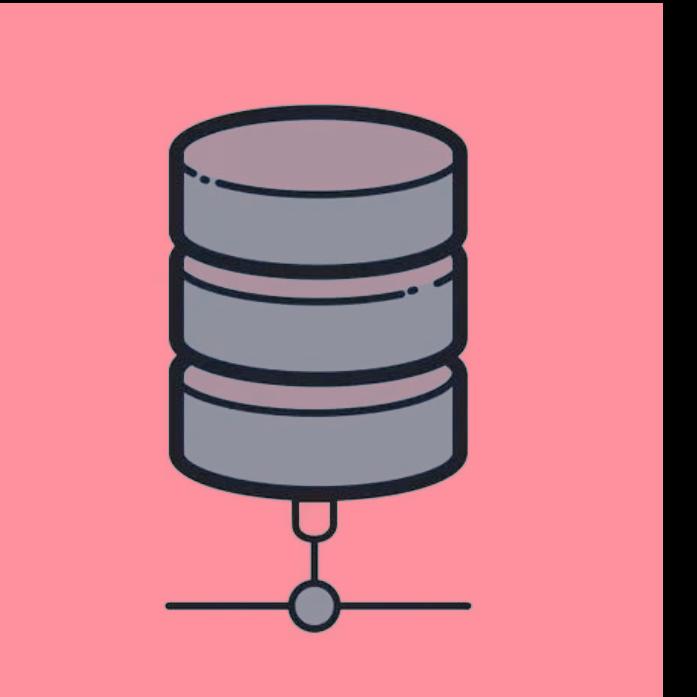
AWS S3 ofrece almacenamiento duradero y escalable, pero puede generar costos ocultos por datos huérfanos, respaldos olvidados o configuraciones inefficientes. FinOps busca visibilidad y control sobre este “desperdicio invisible” que, sin ser detectado, erosiona presupuestos, afecta la eficiencia operativa y revela una débil gobernanza y seguridad en la gestión de datos.

CONCEPTOS BASICOS DE COSTOS EN S3

Que es lo que suma?

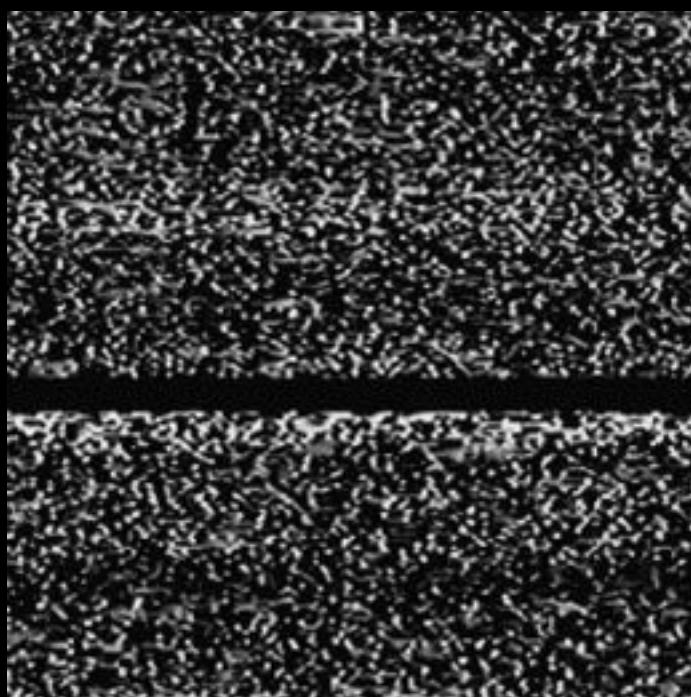
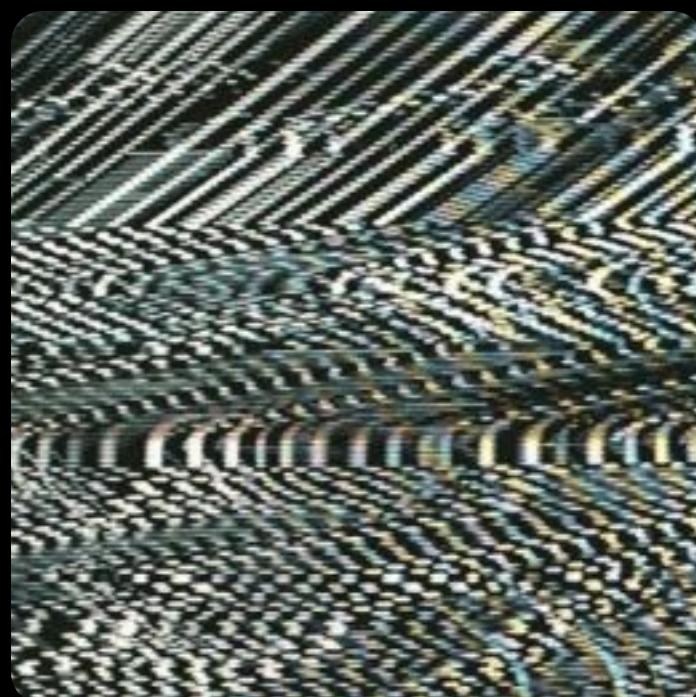
Almacenamiento y duracion del mismo

S3 cobra GB/mes. Mantener data por mas tiempo del necesario implica mayores costos. (2TB de logs en S3 Standard por 1 año es mas caro que simplemente archivarlo en Glacier despues de 30 dias.



Requests y retrievals

Cada PUT/GET/LIST tiene un costo asociado. Clases de acceso infrecuente como Standard-IA, Glacier, cobran por retrieval. El que tanto accedes importa.



Transferencia de datos

Datos salientes entre regiones o a internet generan tarifas de transferencia. Replicas o descargas no planificadas van a aumentar la factura.



Caracteristicas generales

Versionamiento, CRR, MPUs, todo ello puede llevar al incremento en Bytes guardados. Eliminar es gratis.

Standard por 1 año es mas caro que simplemente archivarlo en Glacier despues de 30 dias.

5 EPAM Proprietary & Confidential COSTOS EN S3

<epam>

Requests y retrievals

Que es lo que suma? Cada PUT/GET/LIST tiene un costo asociado. Clases de acceso infrecuente como Standard-IA, Glacier, cobran por retrieval. El que tanto accedes importa.

Transferencia de datos

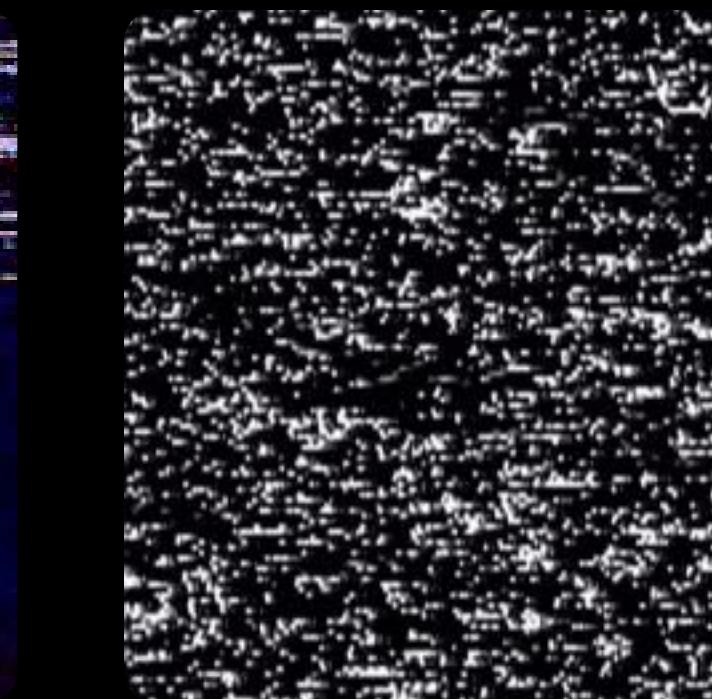
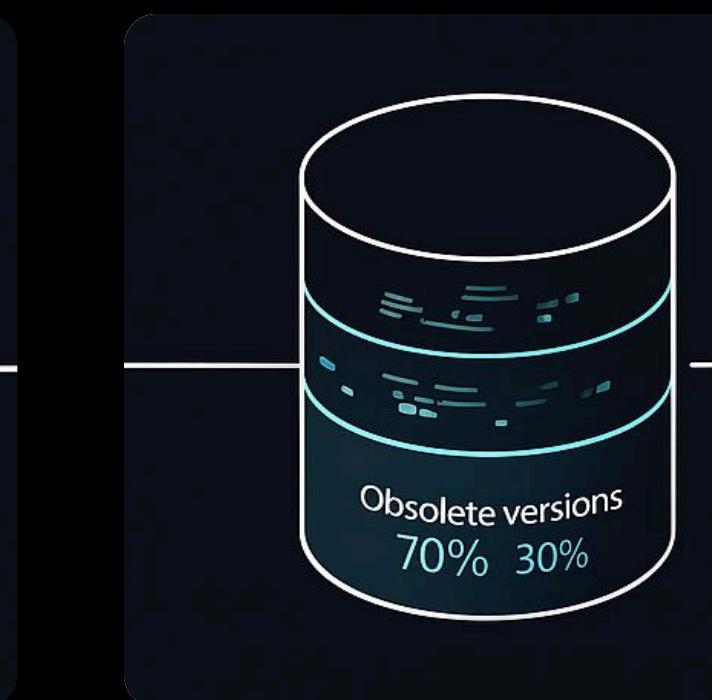
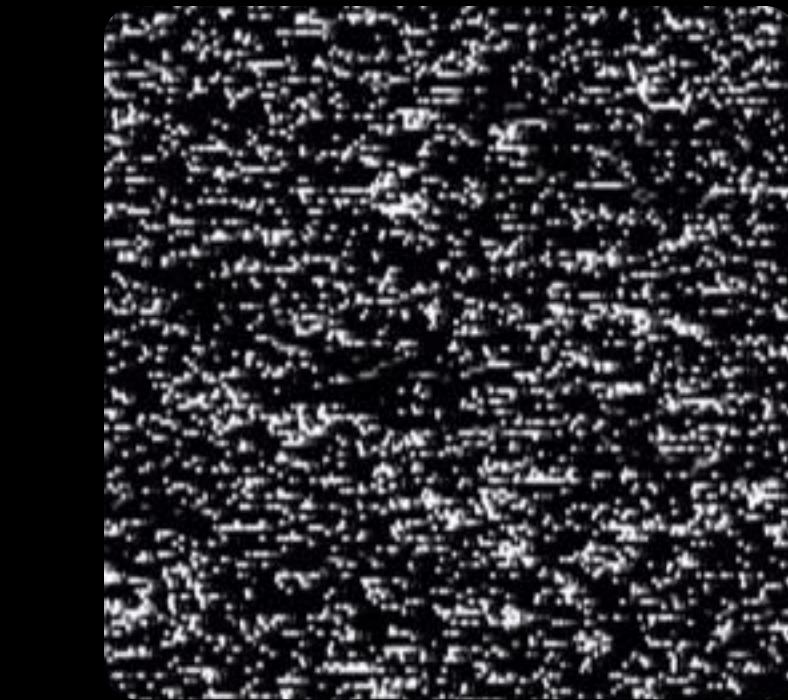
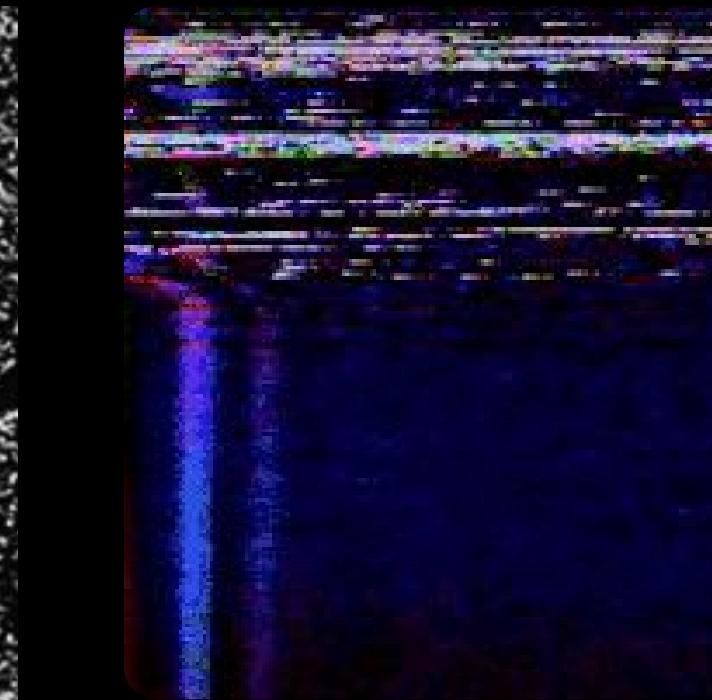
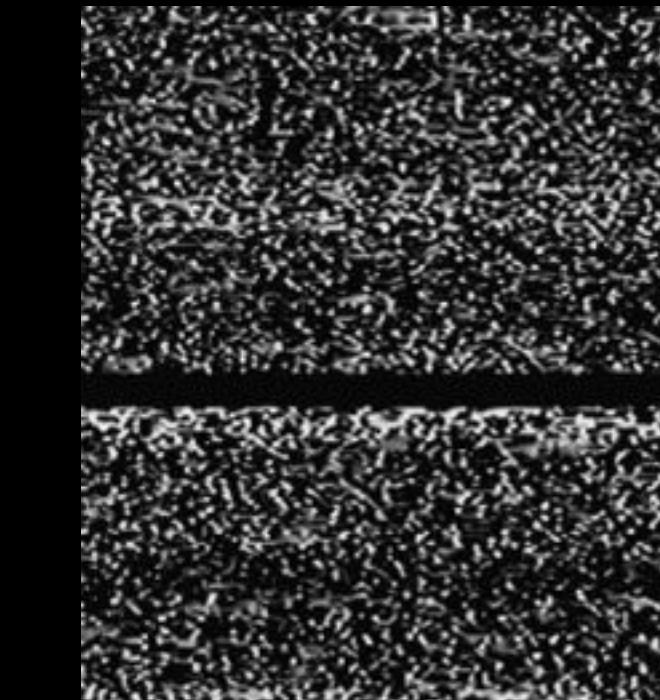
Datos salientes entre regiones o a internet generan tarifas de transferencia. Replicas o descargas no planificadas van a aumentar la factura.

Caracteristicas generales

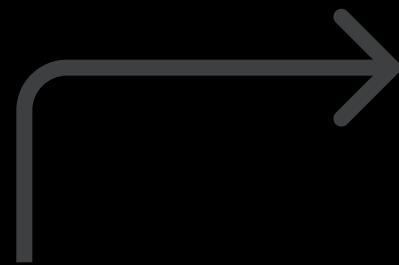
Versionamiento, CRR, MPUs, todo ello puede llevar al incremento en Bytes guardados. Eliminar es gratis pero al usar versionamiento, eliminar solo oculta las lversiones viejas que ocupa espacio e impacta costos

Clases de almacenamiento

Cada clase tiene tarifas diferentes (Standard ~0.023/GB-mes, GDA ~0.004/GB-mes) con un periodo minimo de retenciones. Usar clases de forma incorrecta (guardar backups en standard) implica gastos innecesarios.



Donde se esconde ese desperdicio en S3?



1

Datos huérfanos o innecesarios

Logs antiguos, copias de seguridad vencidas, datos de prueba.

2

Cargas multipartes incompletas

Las cargas multi-partes que no se completan dejan fragmentos almacenados que siguen generando costos hasta que se eliminan.

Pueden representar hasta un 20 % de los costos de S3 si no se controlan

3

Versiones obsoletas

Con versioning habilitado, eliminar o actualizar archivos no borra las versiones antiguas, estas permanecen, consumiendo espacio y dinero.

La importancia de lifecycles.

4

Datos duplicados

Copias redundantes del mismo contenido sean por cargas accidentales, múltiples copias de seguridad o sincronizaciones, entre otras.

Herramientas de deduplicación para identificar objetos idénticos.

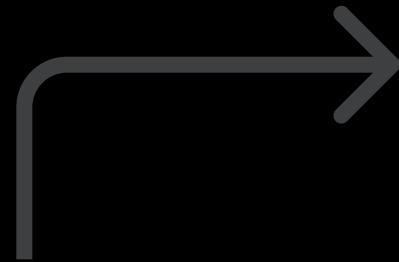
5

Resumen

Costos de S3

Si recuerda

Donde se esconde ese desperdicio en S3?



Cargas multipartes incompletas

Cargas multi-partes que no completan dejan fragmentos incompletados que siguen consumiendo espacio y generando costos hasta que se eliminan.

Representar hasta un tercio de los costos de S3 si no controlan

3

Versiones obsoletas

Con versioning habilitado, eliminar o actualizar archivos no borra las versiones antiguas, estas permanecen, consumiendo espacio y dinero.

La importancia de lifecycles.

4

Datos duplicados

Copias redundantes del mismo contenido sean por cargas accidentales, múltiples copias de seguridad o sincronizaciones, entre otras.

Herramientas de deduplicación para identificar objetos idénticos.

5

Replicacion excesiva y DR innecesario

Copiar datos a múltiples regiones “por si acaso” multiplica los costos de almacenamiento y transferencia.

Si no se requiere alta durabilidad o recuperación ante desastres, esto es bloat invisible



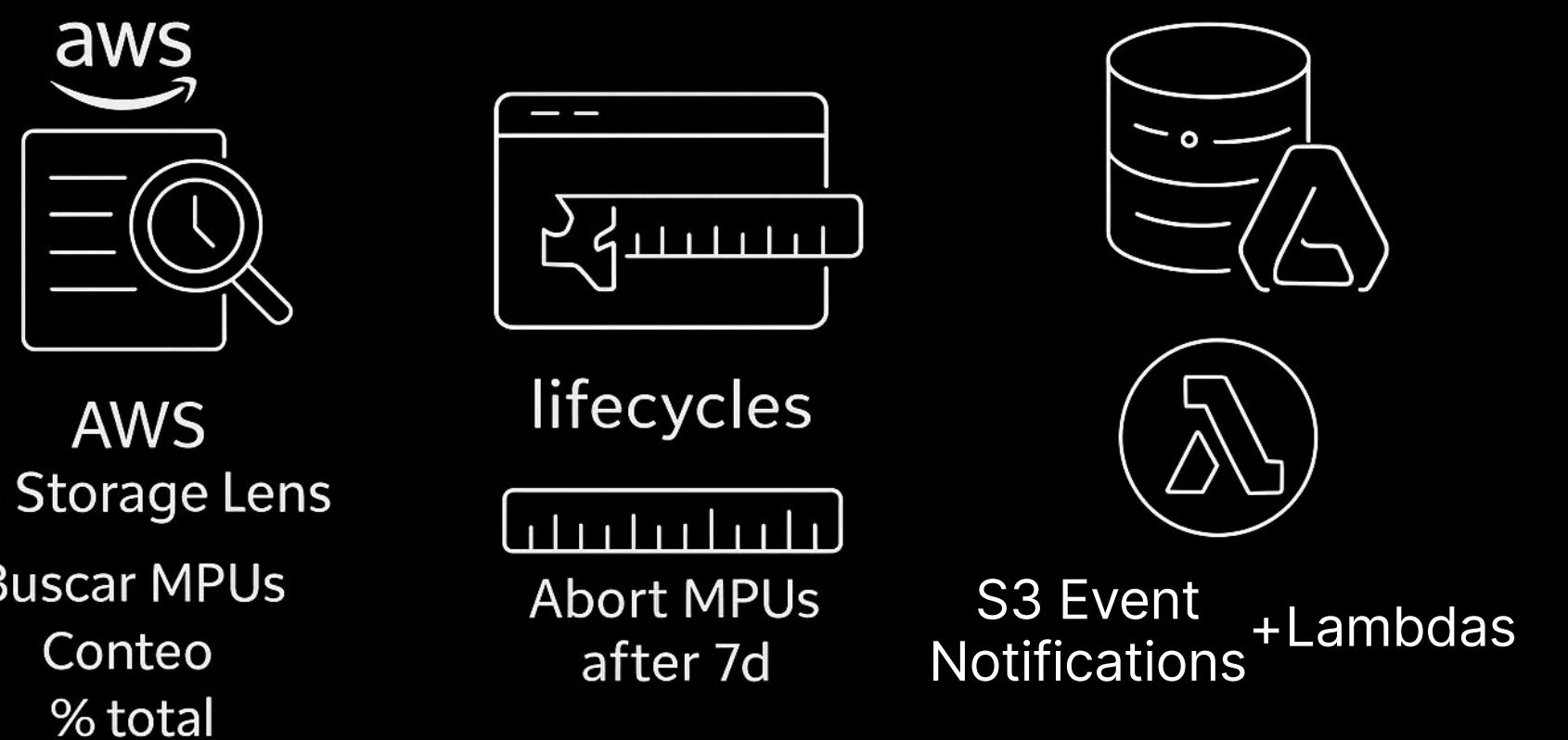
CASO DE ESTUDIO

Multipartes incompletos

Asunto: Una carga multipartes no se completa (sin call “Complete”), las partes subidas permanecen almacenadas indefinidamente. No son visibles en la consola de S3 como archivos, pero se te cobra por ellas hasta que se abortan.

- Habilitar Storage Lens
- Abortar MPUs (7ds)
- Revisa metrica IncompleteMPUsBytes
- Crear alertas en picos
- Inspeccionar top buckets

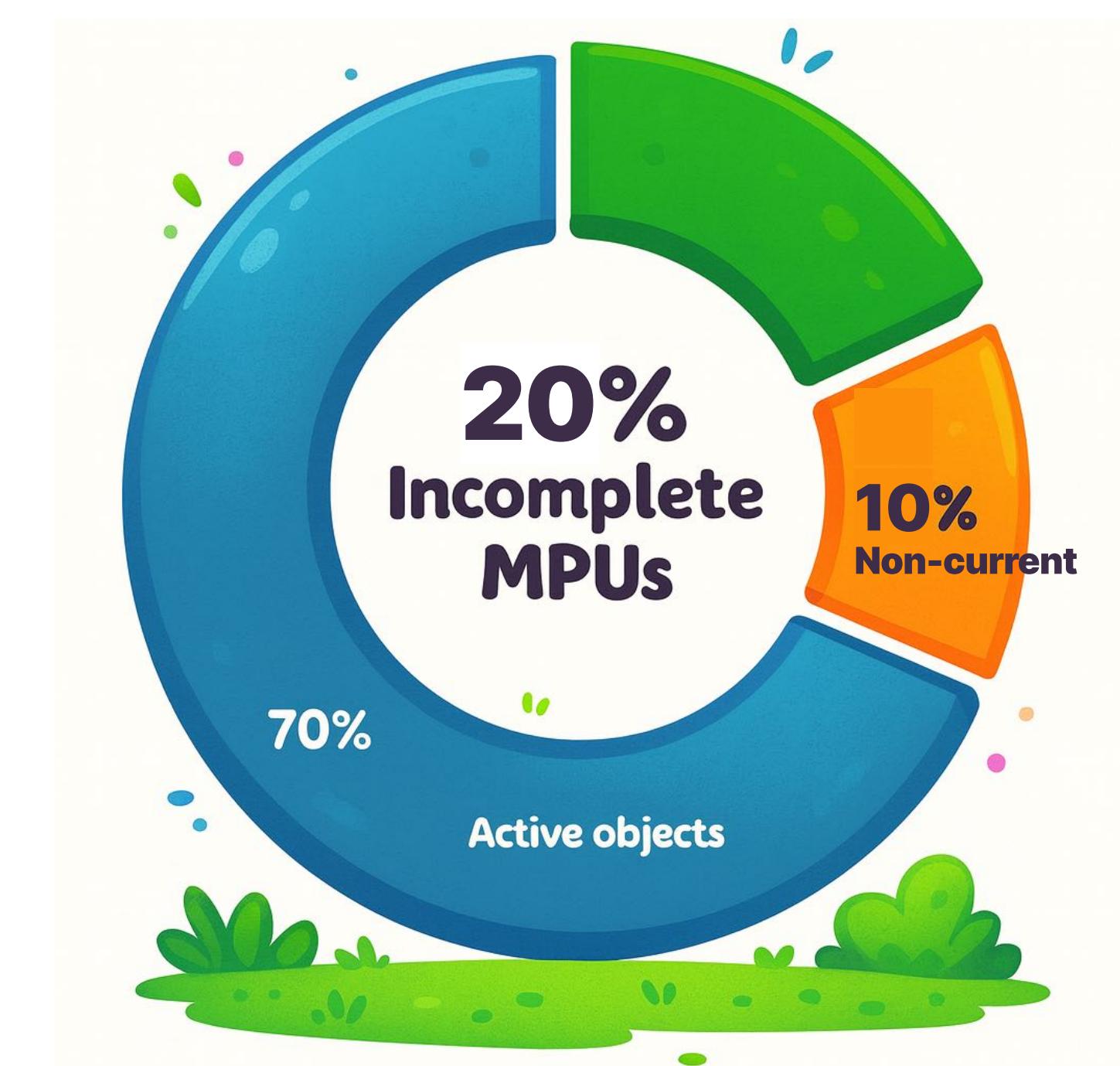
Detección → Remediación → Automatización



IMPACTO

Hasta un ~20%

En el total de lo que estás pagando por tu almacenamiento Puede deberse a cargas incompletas. Con el tiempo las partes huérfanas pueden consumir mucho espacio.



CASO DE ESTUDIO

Proliferación de versiones

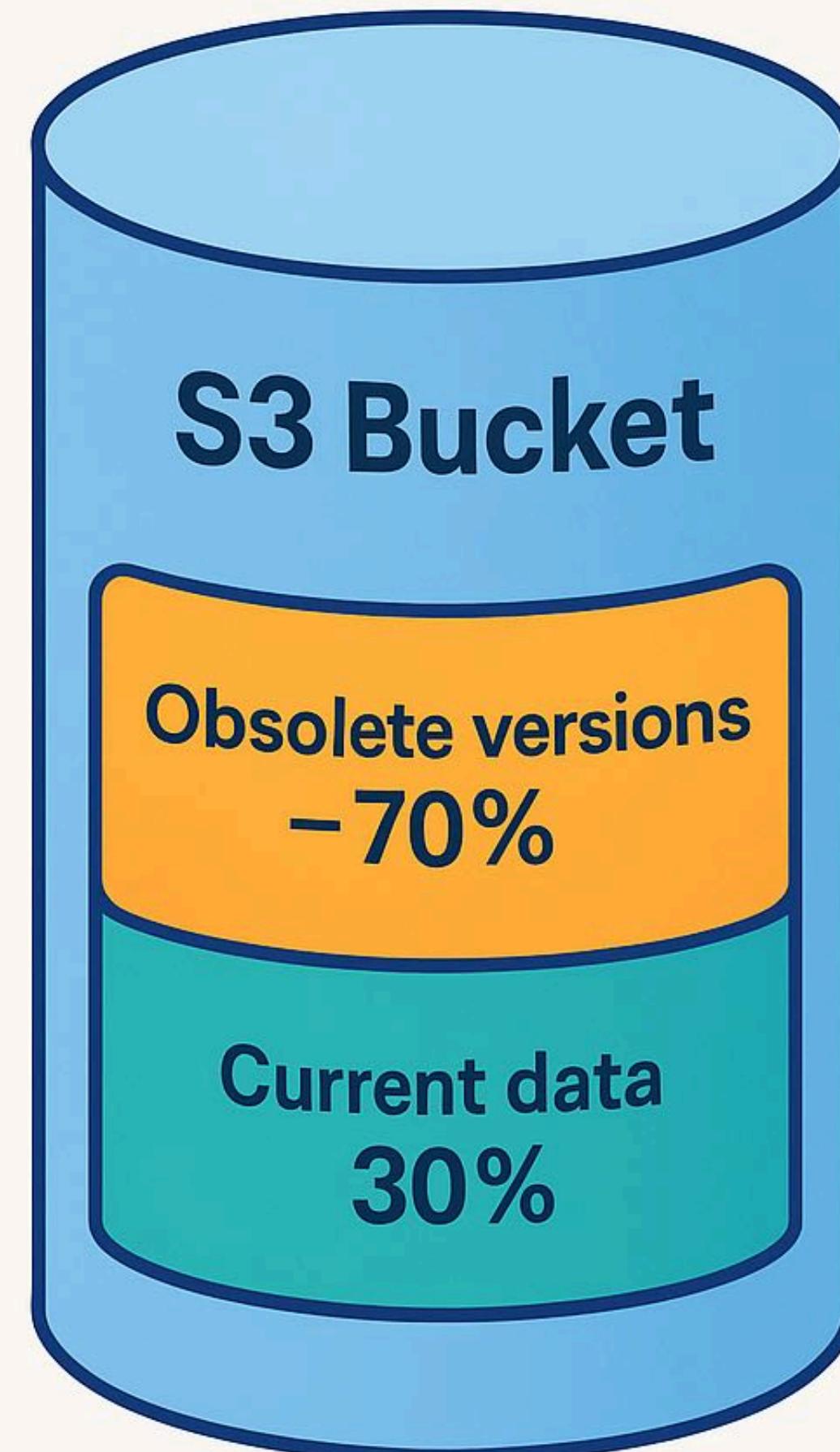
Si bien habilitar el versioning protege contra eliminaciones accidentales, cada versión de archivo cuesta dinero. Esto puede llevar a versiones antiguas que se acumulan indefinidamente.

Uncovering a second cost-saving opportunity

Inspired by the FinOps team's cost investigation, the engineering team dug deeper and discovered an even bigger opportunity to optimize their service's costs. By surfacing both cost data and storage usage in their service dashboards, engineers quickly discovered that the FinOps team's initial hypothesis—that S3 storage classes accounted for the service's cloud costs—wasn't the full story. The engineers came up with a second hypothesis, which focused on the service's use of object versioning, and asked the FinOps team to investigate.

The FinOps team investigated this new theory by visualizing the service's cost data in the [Cloud Cost Explorer](#). They confirmed that, indeed, the service's *primary* cost driver was actually a large number of non-current object versions that S3 retained as a function of its [versioning feature](#). Storage costs could thus be reduced by [minimizing non-current versions](#).

The screenshot shows the Cloud Cost Explorer interface. At the top, it displays a recommendation: "Expire old noncurrent version objects to save \$8890/mo Save 100%". Below this, there are sections for "TOP RECOMMENDATION", "Changes", and "Monthly Cost". The "Changes" section shows a comparison between "Current" and "New" states, with a monthly cost reduction from \$8920 to \$30.



**\$1.5M/-
wasted
on unneeded
versions
(Datadog case)**

CASO DE ESTUDIO

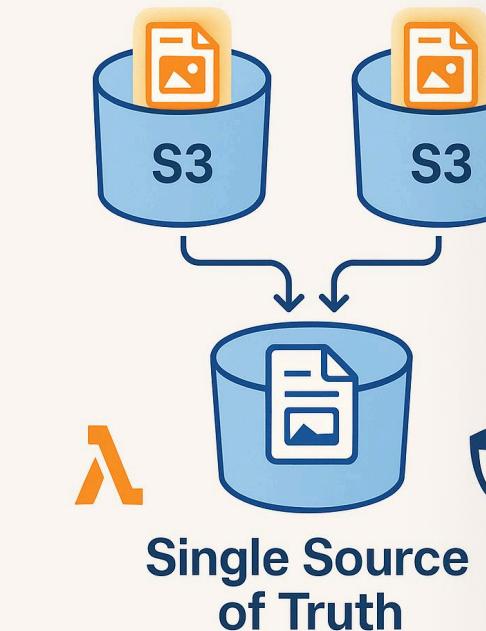
Duplicados

- En data lakes o respaldos, es común encontrar archivos idénticos bajo distintas claves.
- Herramientas como S3 Inventory en conjunto con Athena permiten detectar duplicados por ETag (hash MD5).
- Eliminando copias redundantes con Batch Operations se ahorra espacio y se reduce este tipo de redundancia.

Recomendaciones y buenas prácticas

- Aplica lifecycle rules: expira versiones viejas después de X días.
- Revisa duplicados periódicamente con Inventory + Athena.
- Etiqueta datos con propietario y política de retención.
- Asegura que cada bucket tenga propósito y responsable.

-  **1 Detectar versiones**
-  **2 Aplicar reglas de expiración**
-  **3 Deduplicar**
-  **4 Etiquetar y auditar**



How we saved \$297,400 by deleting duplicate S3 objects AWS

◆ This title was summarized by AI from the post below.



Rewind

27,727 followers
2mo

Tag-and-lifecycle would've cost Rewind \$300,000 to delete duplicate S3 objects.

Instead, our team built a custom solution using Glue, Athena, ECS Fargate, and a Sidekiq-pool cleaner affectionately named Charlie.

 Two weeks for safe marking

 Six days for deletion

 50% reduction in S3 storage for Shopify

 Total cost: \$2,560

This wasn't just a savings win—it was a case study in building observability, respecting rate limits, and making AWS work for you.

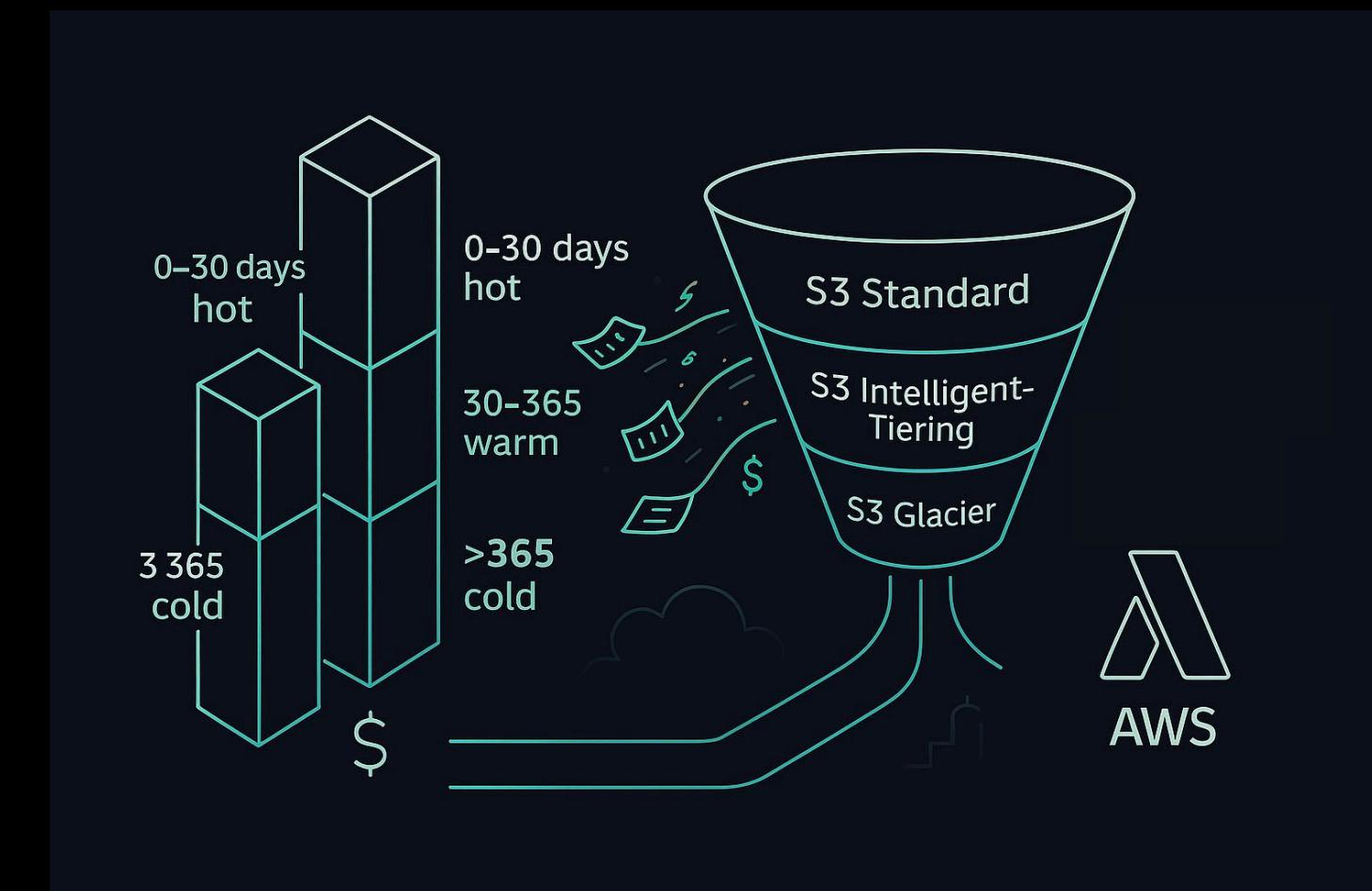
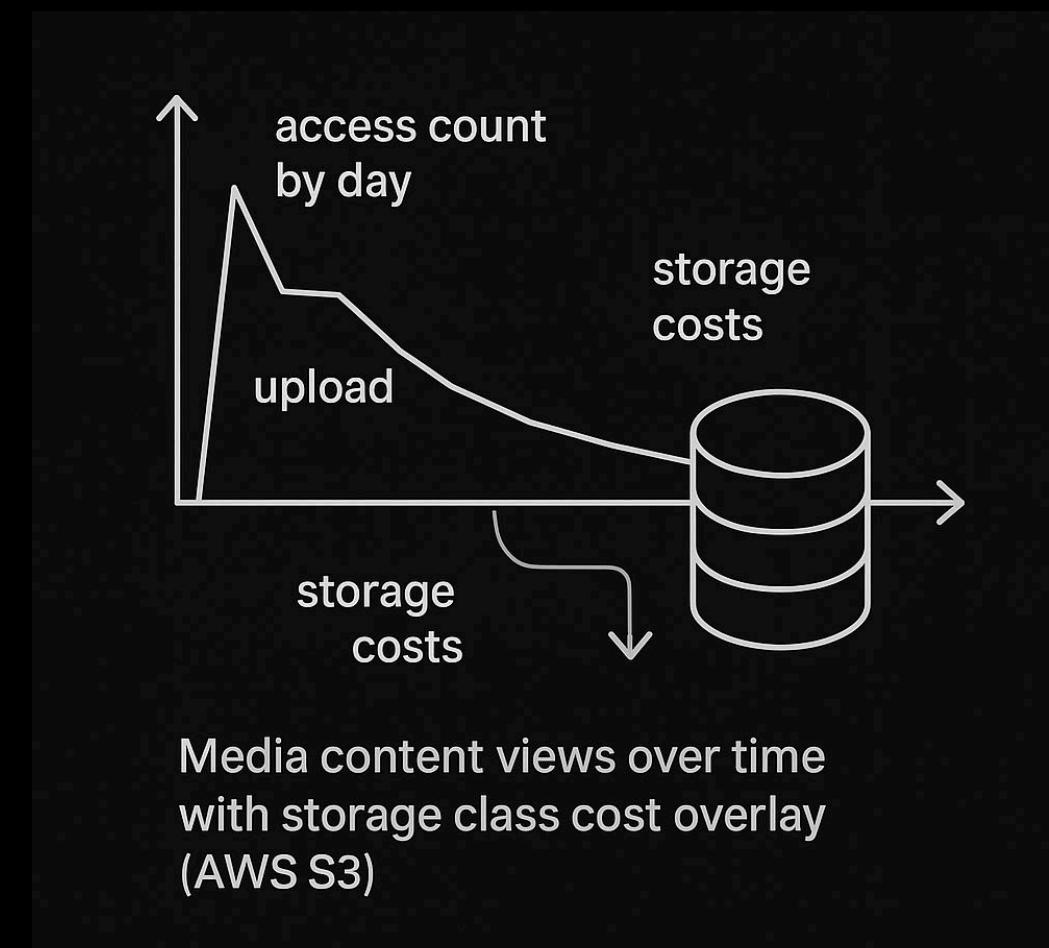
Tech debt doesn't always look like broken code. Sometimes, it's 10 years of nested product integrations.

Read the full write-up here: <https://lnkd.in/erHpPmdC>

#CloudStrategy #FinOps #AWSArchitecture #DataEfficiency #TechDebt #CloudEngineering

From \$300K to \$2.5K

Patrones de acceso de datos - Uno no sirve para todos



Cold Backups

Use case: Nightly backups, escritos una vez y casi nunca leídos.

Patrón de costo: Ideal para Glacier o Deep Archive (almacenamiento ultra-barato, alta latencia de recuperación).

Lifecycle sugerido: Guardar primero en Standard, mover a Glacier después de 1-7 días; eliminar tras el periodo de retención

Hot Content (Streaming / Media)

Use case: Contenido de cara al usuario (imágenes, videos, streaming) con alta demanda tras la carga.

Patrón de costo: Requiere S3 Standard por baja latencia; usar CloudFront para descargar tráfico repetido (reduce costos de data transfer desde S3).

Lifecycle sugerido: Cuando baja la popularidad, mover a Standard-IA o dejar que Intelligent-Tiering lo gestione.

Analytics Data Lakes / Logs

Use case: Grandes volúmenes de logs, sensores, clickstreams; normalmente append-only, consultados ocasionalmente.

Patrón de costo: Data reciente "hot", vieja "warm/cold". Usar Standard-IA para data >30 días, Glacier para >1 año.

Lifecycle sugerido: Vigilar costo de requests; combinar archivos pequeños o usar formatos columnares (Parquet, ORC) reduce overhead.

ELIGIENDO LA CLASE DE ALMACENAMIENTO CORRECTA

Costo vs. Acceso

Storage clas	Cost Tier	Access/Latency	Use-Case	Key Constrain
Standard	Highest	Milliseconds, high throughput	Active, frequently accessed data	If used for cold data → you're overpaying
Standard-IA	~40 % lower than Standard storage cost (source)	Low latency for reads, retrieval fee	Data accessed occasionally (archives, older analytics)	Objects must be >128 KB & stored ≥30 days to be cost-effective
One-Zone-IA	Lower than Standard-IA	Single AZ durability	Non-critical retrievable data	Lower availability & durability
Glacier Instant Retrieval	Much lower storage cost than IA	Millisecond retrieval	"Almost cold" data needing occasional rapid access	Small retrieval fee applies
Glacier Flexible Retrieval	Very low	Minutes-hours retrieval	True archives, backups rarely accessed	Retrieval latency is higher
Glacier Deep Archive	Lowest cost	Hours to retrieve	Long-term retention, rarely accessed	Retrieval latency & minimum duration apply
Intelligent-Tiering	Mid-/auto	Varies (auto transitions)	Data with unpredictable access patterns	Monitoring fee per object; must store ≥30 days

CUANDO USAR CADA CLASE

Una guia breve

- **Contenido activo → Usa Standard.**
Si tus datos se acceden con frecuencia, manténlos en Standard; guardar datos fríos en Standard = dinero desperdiciado.
- **Acceso ocasional → Usa Standard-IA.**
Ideal para archivos de usuario o datos analíticos antiguos; requiere objetos de más de 128 KB y almacenarlos al menos 30 días para que sea rentable.
- **Recuperable pero no crítico → Usa One-Zone-IA.**
Más barato que IA; adecuado si aceptas menor durabilidad (una sola zona de disponibilidad).
- **Casi frío pero con accesos rápidos ocasionales → Usa Glacier Instant Retrieval.**
Bajo costo y aún con recuperación rápida cuando se necesita (cumplimiento normativo, auditorías).
- **Archivos fríos → Usa Glacier Flexible Retrieval o Deep Archive.**
Si el acceso es muy raro y la latencia no importa, Deep Archive ofrece el costo más bajo.
- **Patrones inciertos o cambiantes → Usa Intelligent-Tiering.**
Mejor opción cuando no conoces el patrón de acceso; mueve los objetos automáticamente entre niveles. Pero tiene un pequeño costo de monitoreo, así que asegúrate de que el tamaño y tiempo de vida de los objetos lo justifiquen.

NO PUEDES OPTIMIZAR LO QUE NO PUEDES VER

Herramientas para visibilidad en S3

AWS S3 Storage Lens

Qué es: Panel analítico con +60 métricas de uso y costos.

Uso: Detecta buckets grandes, sin reglas de ciclo de vida o con cargas incompletas.

Idea visual: Mapa de calor de buckets con alertas de “waste”.

CloudWatch Metrics

Qué es: Métricas y alarmas sobre tamaño, objetos y actividad.

Uso: Alerta si un bucket crece más de X GB en un día.

Idea visual: Icono de alarma sobre un bucket.

AWS Cost Explorer

Qué es: Visualizador de costos por servicio, región o bucket.

Uso: Identifica saltos inusuales en costos o transferencias.

Idea visual: Gráfica de barras mostrando “pico de gasto S3”.

S3 Inventory + Athena

Qué es: Reporte CSV/Parquet con todos los objetos.

Uso: Detecta duplicados (por ETag), datos viejos o grandes.

Idea visual: Tabla + lupa con íconos de duplicados.

Storage Class Analysis

Qué es: Analiza accesos para sugerir clases más baratas.

Uso: “Objetos sin acceso en 90 días → mover a IA.”

Idea visual: Flecha de transición “Standard → IA”.

Estas herramientas permiten descubrir el ‘desperdicio invisible’ y priorizar acciones de optimización.

STUDY CASE

Regla de Lifecycle en S3 que costó \$690,000

Un ingeniero activó una regla de archivado en S3 para “ahorrar costos”.

6.9 mil millones de objetos de ~10 KB fueron movidos automáticamente a Glacier Deep Archive.

Resultado: la factura aumentó en lugar de disminuir.

6.9G Objetos de ~10KB

fueron movidos automáticamente a Glacier Deep Archive.



Qué salió mal

- Costo por transición:** cada objeto genera un cargo individual por solicitud, no por GB.
- Objetos pequeños:** el tamaño mínimo de facturación fue 128 KB, así que cada archivo de 10 KB se cobró como 128 KB.
- Duración mínima:** costos de retención aplicaron incluso si los objetos eran recientes.

Riesgo	Control FinOps Preventivo
Transición masiva sin análisis	Revisión de Lifecycle Rules (FinOps Policy #04): análisis de impacto antes del despliegue.
Falta de visibilidad sobre tamaño/volumen	S3 Inventory + Cost Simulation Tool para estimar costo por transición.
Automatización sin pruebas	Prueba del 1 % antes del rollout global.
Falta de guía sobre objetos pequeños	Threshold Policy: no archivar objetos menores a 128 KB.



Mover datos pequeños masivamente puede ser más caro que mantenerlos.



Referencia LinkedIn



Post original de Mikael Almstedt

CLAVES PARA OPTIMIZAR S3

Costo y Eficiencia



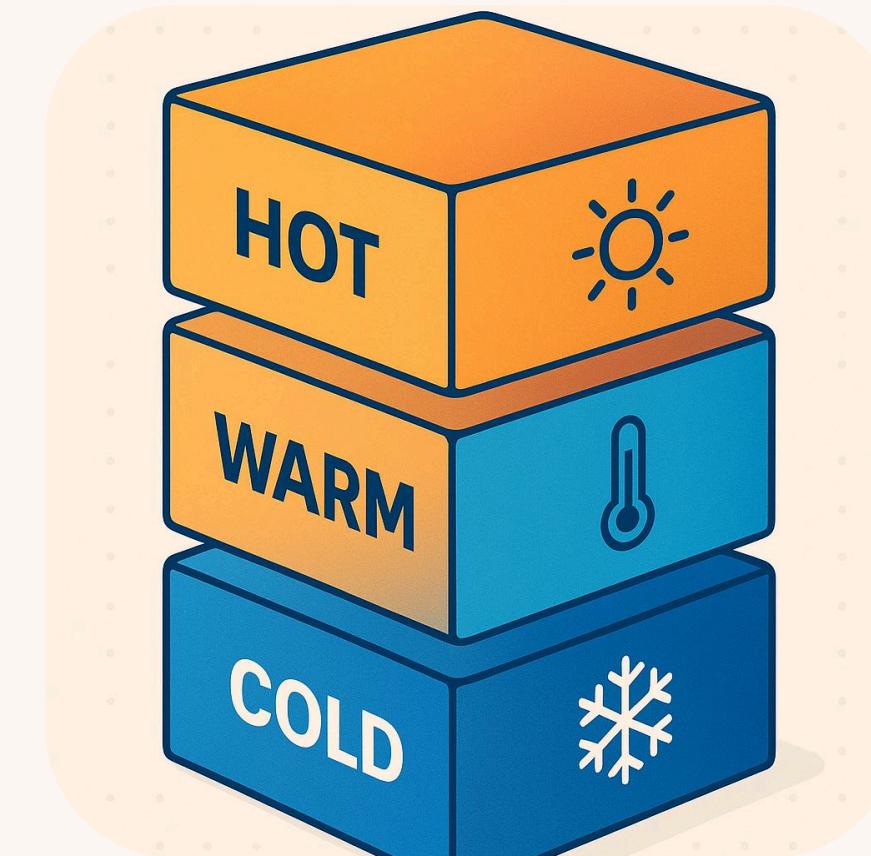
Borra o archiva lo innecesario

El byte más barato es el que no se guarda. Revisa y elimina respaldos, logs o duplicados antiguos



Automatiza con Lifecycle Policies

Aplica reglas de ciclo de vida: transfiere datos, expira versiones y aborta cargas incompletas.



Usa la clase correcta

Ajusta la clase de almacenamiento según el uso: Standard, IA o Glacier. Evita pagar de más por inercia.



Activa visibilidad y alertas

Usa Storage Lens, Cost Explorer y alertas para detectar anomalías antes de que crezcan.



Aplica prácticas FinOps

Haz del ahorro una operación continua. Revisa costos, define políticas y mide resultados.

Gracias!

Referencias

- Amazon S3 “Storage Classes” documentation — gives durability, retrieval fees, min-duration, etc. [Amazon Web Services, Inc.+1](#)
- FinOps Foundation / Working Group on S3 Cost Optimization — “Intelligent-Tiering is the only storage class that delivers automatic cost savings... when access patterns change.” [FinOps Foundation+1](#)
- Blog/white-paper: “Maximizing AWS S3 Cost Efficiency...” — emphasises the trade-off between cost & latency and importance of aligning class to pattern. [cloudride.co.il](#)

