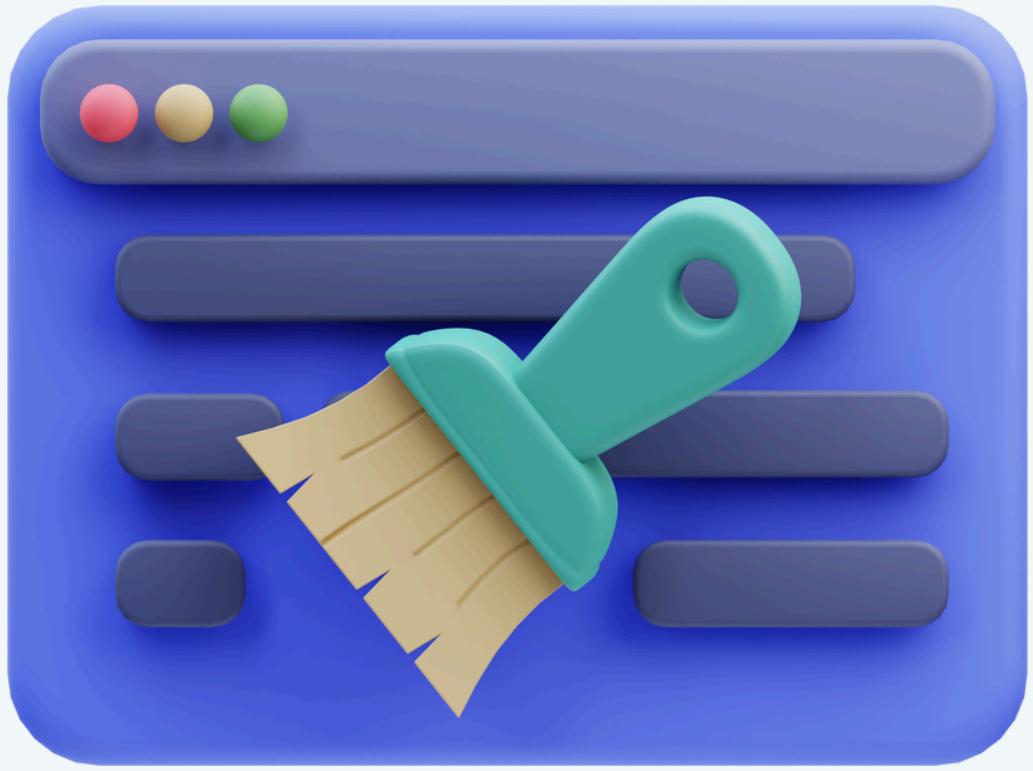
Cacheo de Datos

La guía definitiva, con las principales técnicas para hacer tus aplicaciones mucho más rápidas y efectivas



PABLO DEL ÁLAMO

Introducción

El caching es una técnica que guarda datos en una memoria rápida.

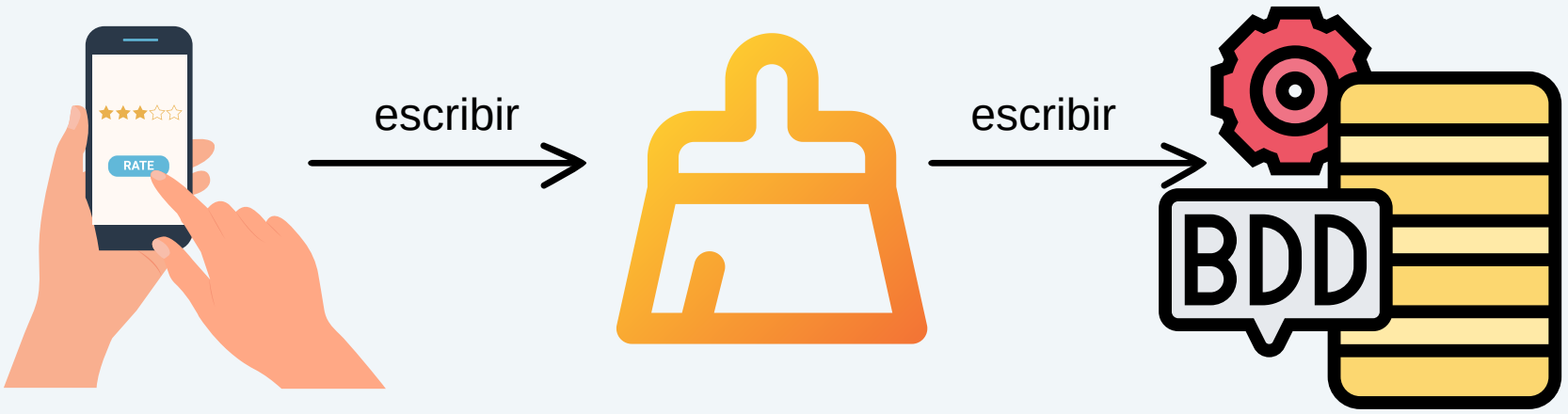
> Así, cuando vuelvas a necesitar esos datos... ¡ya estarán ahí! No necesitas ir a buscarlos al origen de nuevo. ⏩

Usamos caching para mejorar la velocidad y eficiencia de nuestras aplicaciones. Menos consultas a la base de datos significa menos esperas y una mejor experiencia de usuario.

PABLO DEL ÁLAMO

Write-Through: Concepto

El método Write-Through escribe datos simultáneamente en la caché y en la base de datos. Así, ¡todo está siempre sincronizado! 🙌



PABLO DEL ÁLAMO

Ventajas del Write-Through

Consistencia asegurada: al escribir en el caché y la base de datos a la vez, se minimizan las discrepancias.

Siempre tendrás la versión más actualizada de los datos.

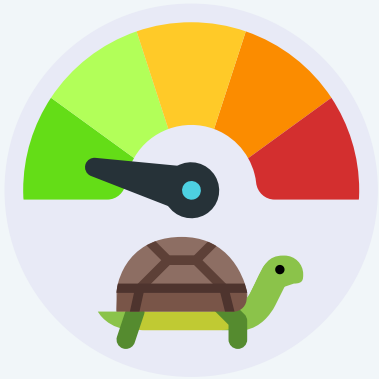
PABLO DEL ÁLAMO

Desventajas del Write-Through

El coste de la consistencia es la velocidad.

Al escribir en dos lugares a la vez, las operaciones pueden ser más lentas.

Mucha seguridad, pero menos velocidad.

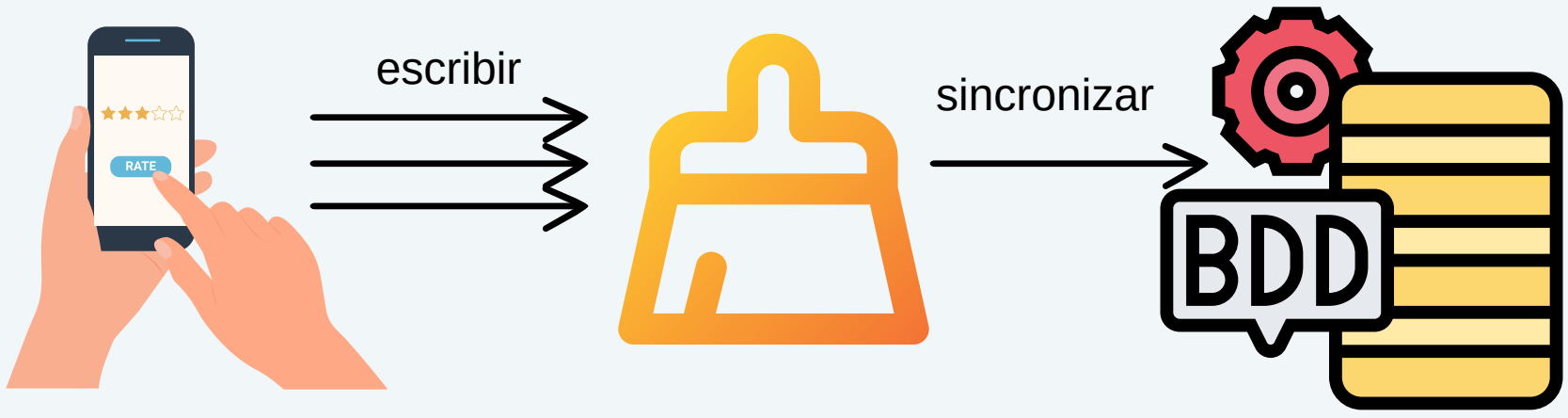


PABLO DEL ÁLAMO

Write-Back: Concepto

Write-Back es como un refugio rápido.

Guardas inicialmente en la caché y luego, en un momento dado, haces flush a la base de datos.



PABLO DEL ÁLAMO

Ventajas del Write-Back

Ganas velocidad porque no necesitas escribir inmediatamente en la base de datos.

Ideal para operaciones donde la rapidez inicial es crucial.

PABLO DEL ÁLAMO

Desventajas del Write-Back

Con la velocidad vienen riesgos.

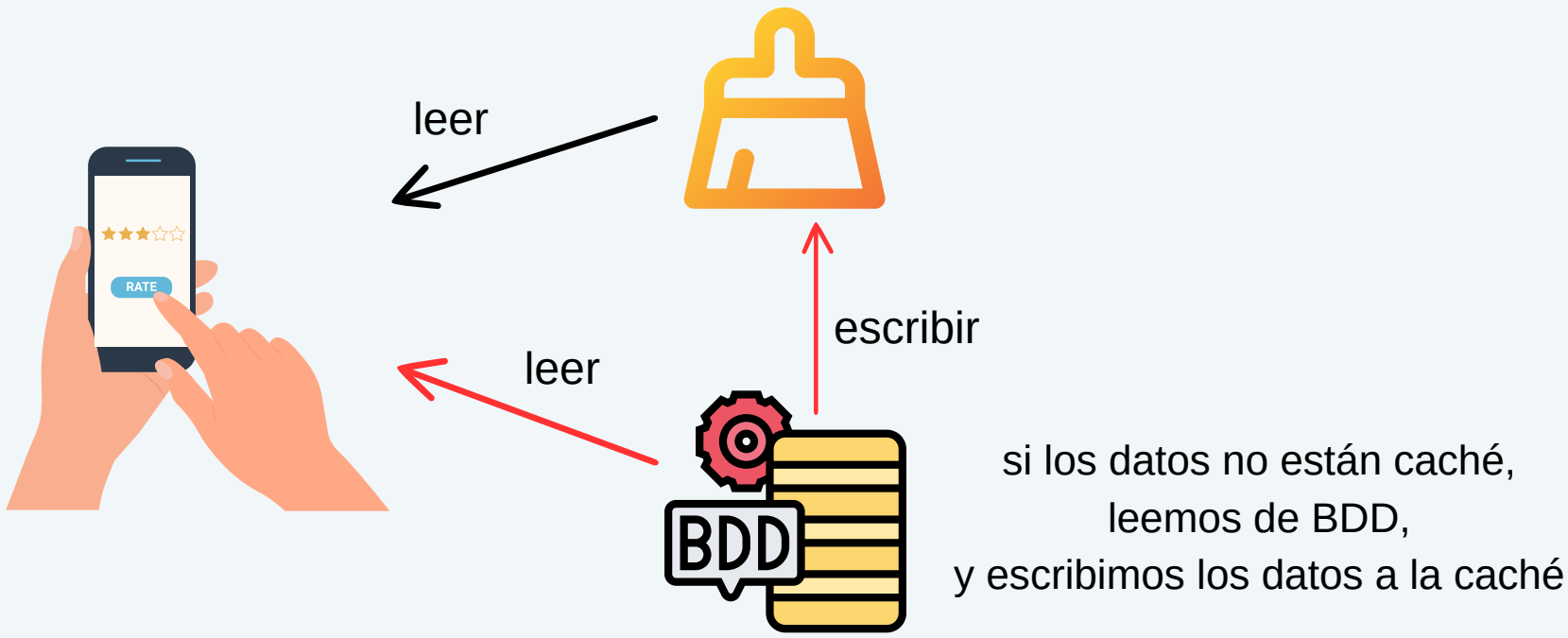
Si la caché falla antes de actualizarse la base de datos, puedes perder datos.

PABLO DEL ÁLAMO

Cache-Aside (Lazy Loading): Concepto

Cache-Aside o Lazy Loading es una técnica donde cargas datos en la caché cuando se solicitan.

Si no están, se obtienen de la base de datos.

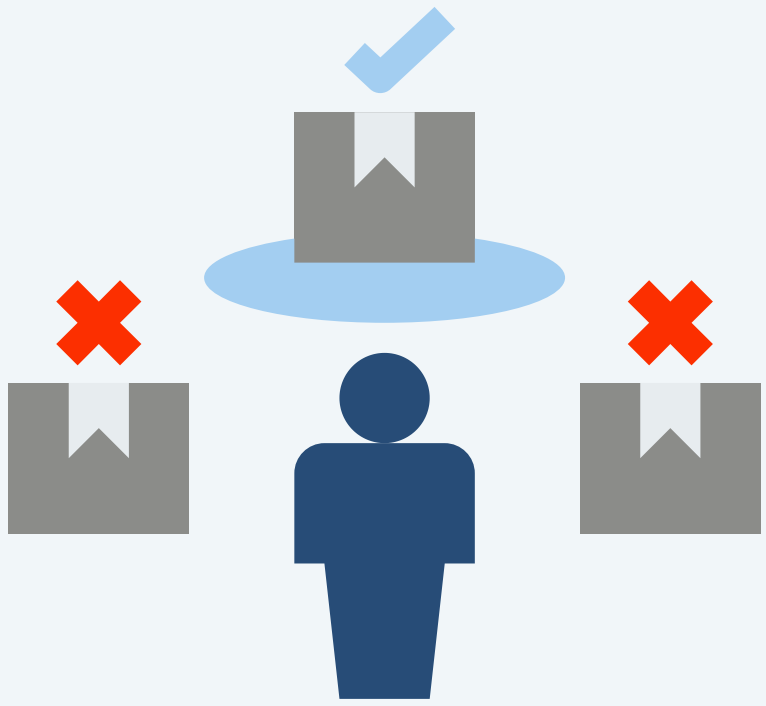


PABLO DEL ÁLAMO

Ventajas del Cache-Aside

Eficiencia: solo cacheas los datos necesarios.

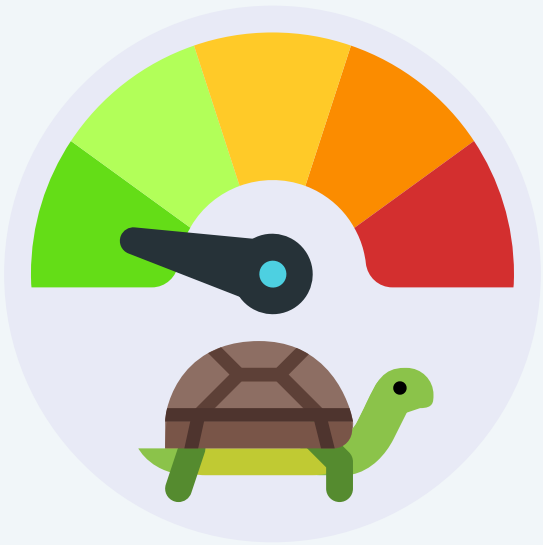
Reduces la sobrecarga y evitas llenar la caché con datos inútiles.



PABLO DEL ÁLAMO

Desventajas del Cache-Aside

El primer acceso a datos no cacheados puede ser lento, ya que necesita rescatar la información de la base de datos y cargarla en la caché.



PABLO DEL ÁLAMO

Write-Around: Concepto

Write-Around es una estrategia donde las operaciones de escritura van directamente a la base de datos, evitando la caché completamente en esa etapa.

La memoria caché solo se actualiza si se vuelven a leer los mismos datos.

PABLO DEL ÁLAMO

Ventajas del Write-Around

Solucionas el problema de datos obsoletos en la caché al no escribir directamente en él.

Esto previene lecturas inconsistentes.

PABLO DEL ÁLAMO

Desventajas del Write-Around

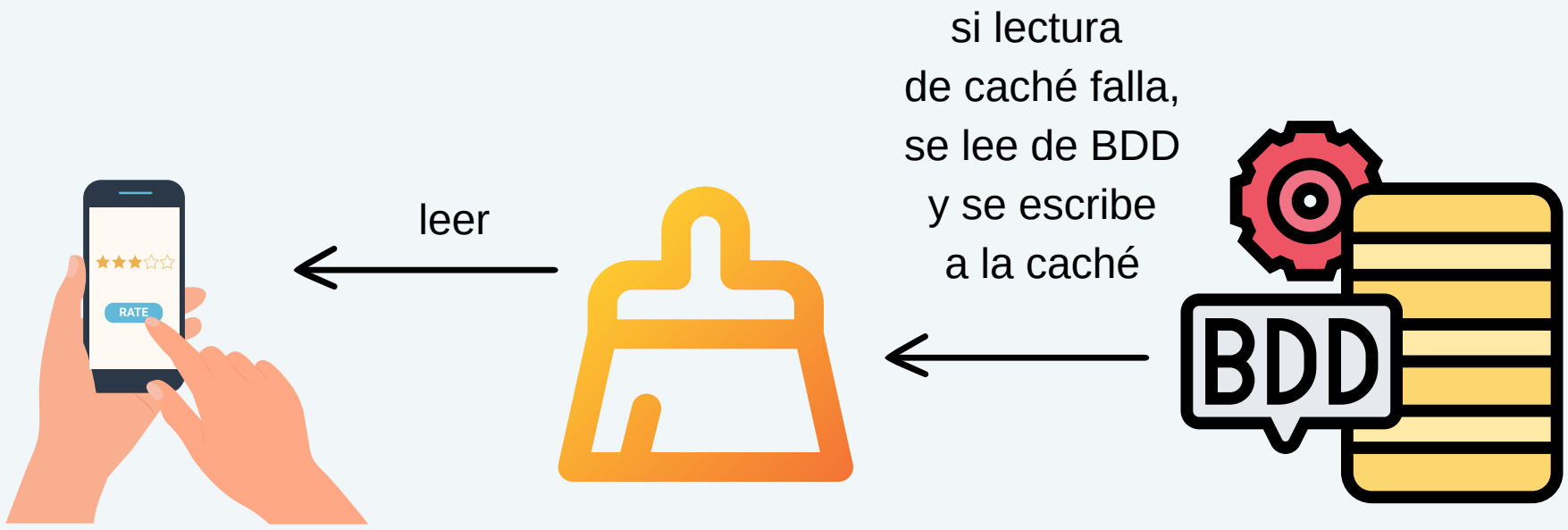
Los datos recién escritos no están inmediatamente disponibles en caché.

La primera lectura después de una escritura irá a la base de datos.

PABLO DEL ÁLAMO

Read-Through: Concepto

Read-Through es cuando la caché maneja automáticamente las solicitudes de datos, obteniéndolos de la base de datos si no están ya cacheados.



PABLO DEL ÁLAMO

Ventajas del Read-Through

El cliente interactúa únicamente con la caché, dejando que esta gestione automáticamente los cache misses.

Esto reduce la complejidad del código y centraliza la lógica de acceso a datos.

PABLO DEL ÁLAMO

Desventajas del Read-Through

Como la caché decide cuándo obtener datos, puede que no siempre se optimice el consumo de datos.

PABLO DEL ÁLAMO

Refresh-Ahead: Concepto

Refresh-Ahead actualiza los datos de la caché antes de que caduquen.

Esto se anticipa a llamadas frecuentes de datos, manteniendo la caché actualizada.

PABLO DEL ÁLAMO

Ventajas del Refresh-Ahead

Esta estrategia reduce al mínimo los tiempos de espera causados por la expiración de la caché, manteniendo la información accesible.

PABLO DEL ÁLAMO

Desventajas del Refresh-Ahead

Si refrescas sobre datos que no se usan, puedes estar gastando recursos innecesariamente.

Estrategia útil, pero cuidado al aplicarla.

PABLO DEL ÁLAMO

Elegir la Técnica de Caché Adecuada

Como siempre, cada caso de uso es un mundo, y tu rol como dev/ingenier@ es saber determinar qué técnica usar dependiendo de la situación.

Aún así, te dejo una serie de casos de uso y pautas, para guiarte en cuándo usar cada técnica:

- Write-Through: Úsalo cuando priorices la consistencia sobre la velocidad. Ideal para sistemas donde perder datos es un serio problema, como registros financieros. 💰

PABLO DEL ÁLAMO

- Write-Back: Perfecto si necesitas rapidez para operaciones intensivas donde una pérdida ocasional de datos no sea catastrófica, como en buffers de video. 📹

- Cache-Aside (Lazy Loading): Opta por esta técnica cuando desees flexibilidad y eficiencia, solo guardando datos frecuentemente accedidos. Útil en aplicaciones de contenido esporádico. 📚

PABLO DEL ÁLAMO

- Write-Around: Si te preocupa la integridad de datos durante escrituras frecuentes, pero las lecturas son esporádicas, esta es tu técnica. Ejemplo: registros de sesión. 💾

- Read-Through: Indicada para simplificar el código donde se necesita acceso constante y rápido a los datos, como catálogos de productos. 🛍️

- Refresh-Ahead: Perfecto para cachés con alta frecuencia de acceso donde los datos deben estar actualizados, como estadísticas en tiempo real. 📊

PABLO DEL ÁLAMO

Conclusión

Tras haber visto estas diapositivas, deberías tener una visión global de diversas técnicas de caching, cada una con sus puntos fuertes y aplicaciones óptimas.

El caching es una herramienta poderosa para optimizar el rendimiento de tus aplicaciones y proporcionar una experiencia de usuario mejorada.

Recuerda que no existe un enfoque único que funcione para todo, así que evalúa cuidadosamente tus requisitos específicos, como consistencia, velocidad y frecuencia de acceso a los datos.

¡Experimenta, ajusta y encuentra la combinación perfecta para llevar tus proyectos al siguiente nivel!

PABLO DEL ÁLAMO

¿Te ha resultado útil?

- Comparte esta guía con tu equipo o amigos desarrolladores.
- Guárdala para tenerla siempre a mano.
- ¡Dale un like o comenta si tienes preguntas!

PABLO DEL ÁLAMO