

# Modelos de consistencia de MEMORIA



Equipo:

**Villalba Gonzalez Ariana Vanesa**  
**Beltran Zazueta Emily**  
**Machado Félix Flor María**  
**Mendoza Moreno Mariana**

# Introducción

Un modelo de consistencia de memoria es el conjunto de reglas que gobierna como los sistemas de memoria procesarán las operaciones de memoria (carga/almacenamiento) desde **múltiples procesadores.**



## Por qué son necesarios estos modelos?

Por que en los sistemas modernos hay multiples CPUs y las operaciones pueden ejecutarse fuera de orden. Esto provoca que dos procesos vean datos distintos y el orden de las operaciones no sea el esperado.

# Conceptos basicos

## Operaciones

- **Read (lectura)**

Obtiene el valor de memoria

- **Write (escritura)**

modificar el valor

## Orden de operaciones

- **Orden del programa**

Orden de las instrucciones en el codigo

- **Orden Observado**

Orden en que otros procesadores perciven las operaciones

# Modelos de consistencia FUERTE

## Características generales:

- Orden secuencial estricto
- Simplicidad
- Vicibilidad instantánea
- Determinismo

# Consistencia Estricta

Garantiza que cualquier operación de lectura sobre una variable siempre devolverá el valor de la escritura más reciente en tiempo real, Es el modelo tradicionalmente proporcionado por los uniprocesadores y es con el cual se escriben la mayoría de los programas.

No es realista para sistemas multiprocesador ya que no tiene en cuenta la latencia de comunicación.

# Consistencia Secuencial

Garantiza que exista una única secuencia, globalmente observable, de operaciones de carga, almacenamiento y lectura, modificación, escritura en cualquier ejecución válida. Recibe este nombre porque da la impresión de que todo el programa se ejecuta mediante un procesador secuencial simple y ordenado que realiza múltiples tareas entre los hilos disponibles.

Resultado de una ejecución es el mismo que el de una única intercalación de accesos secuenciales, en orden de programa, desde diferentes procesadores.

# Consistencia Causal

Es uno de los modelos mas utilizados en sistemas distribuidos porque garantiza que las operaciones de lectura y escritura relacionadas logicamente como causa y efecto se observen siempre en el mismo orden en todos los nodos y operaciones que no estan relacionadas sean en un orden distinto y asincrono.

Las operaciones relacionadas causalmente deben aparecer en el mismo orden en todos los procesos, aunque estos puedan discrepar sobre el orden de las operaciones causalmente independientes.

# Consistencia Linealizabile

Es un modelo en sistemas concurrentes y distribuidos donde todas las operaciones parecen ejecutarse instantaneamente en un unico momento exacto.

Por lo que se obtiene un orden en tiempo real ya que si una operacion termina antes que otra el sistema asegura que la segunda nunca se mostrara antes.

Es considerado como un sistema de alta latencia y baja disponibilidad, ya que se pueden generar cuellos de botella en operaciones de escritura y lectura. Pero es de alta confiabilidad y de sencilla trazabilidad.

# Modelos de consistencia DEBILES

## Características generales

- Propagación retardada
- Falta de Sincronización
- Posibles lecturas obsoletas
- Convergencia eventual no garantizada

# Consistencia débil

La concistencia débil es un modelo de consistencia en sistemas distribuidos que ofrece la garantía más baja respecto a la sincronización de datos entre múltiples réplicas. En un sistema de consistencia débil no existen requisitos estrictos sobre cuándo o si las réplicas convergerán a un estado consistente.

La consistencia débil permite una divergencia significativa entre las réplicas y admite inconsistencias temporales

# Consistencia de Liberación

- Es más relajada que al consistencia débil.
- Accesos de sincronización divididos en:
  - Acquire → Adquisición
  - Release → Liberación

**Acquire:** Debe completarse antes que todos los accesos subsiguientes.

**Release:**

- Deben completarse todos los accesos de memoria previos.
- Accesos de memoria posteriores SI pueden iniciarse.
- Opreaciones que siguen a release y que deben esperar se deben proteger con un acquire

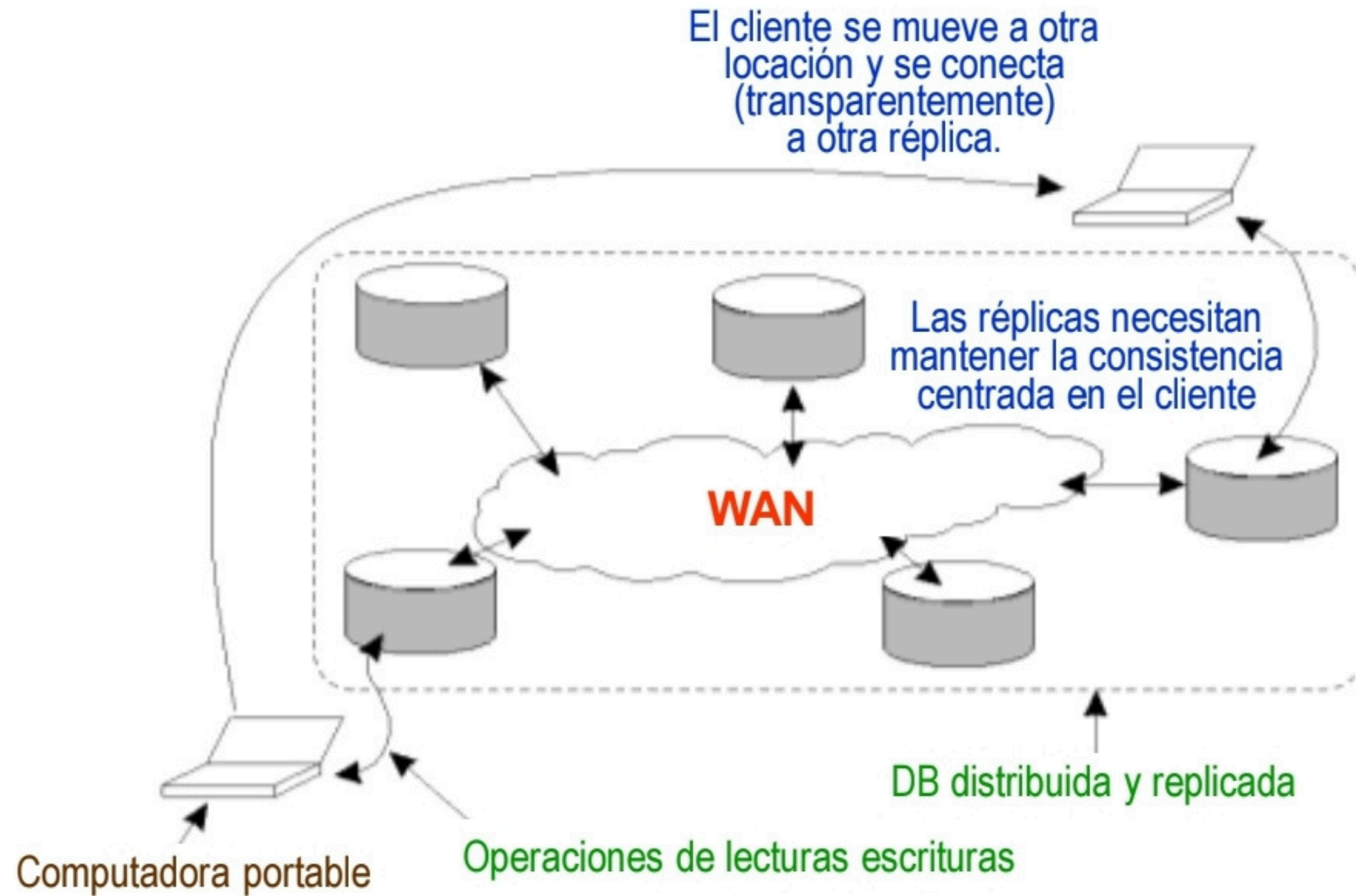


# Consistencia Eventual

Se trata de un modelo de consistencia utilizado en sistemas distribuidos donde las actualizaciones de datos se propagan a todos los nodos de la red a lo largo del tiempo, en lugar de instantáneamente. Este modelo permite que el sistema continúe operando ante fallos parciales o particiones de red, priorizando así la disponibilidad y la tolerancia a particiones sobre la consistencia inmediata.

- **Alta disponibilidad** : Los sistemas pueden seguir funcionando y aceptando actualizaciones incluso cuando algunos nodos están fuera de línea.
- **Rendimiento mejorado** : Ofrece un mejor rendimiento y una menor latencia en las operaciones de escritura.
- **Escalabilidad** : El sistema gestione más transacciones simultáneamente.

# Consistencia Eventual



# Comparación

## Características

## Modelo Fuerte

## Modelo Debil

**Facilidad de programación**

Alta

Baja

**Rendimiento**

Menor

Mayor

**Escalabilidad**

Limitada

Alta

**Consistencia Inmediata**

Sí

No siempre

**Complejidad de hardware**

Alta

Menor

**Uso moderno**

Limitado

Muy común

# Aplicaciones

- La arquitectura Intel x86 utiliza un modelo fuerte de consistencia de memoria denominado Total Store Ordering (TSO).
- La arquitectura ARM utiliza un modelo de consistencia de memoria débil llamado weakly ordered memory model (WOMM).
- La arquitectura NVIDIA CUDA utiliza un modelo de consistencia de memoria débil llamado consistencia de ámbito.

# Conclusión

- Los modelos de consistencia de memoria definen como los procesos observan y comparten datos en sistemas concurrentes y distribuidos.
- Los modelos fuertes ofrecen mayor coherencia y facilidad de programación pero reducen el rendimiento y la escalabilidad.
- Los modelos débiles permiten mayor desempeño, sistemas distribuidos masivos pero aumentan la complejidad.

**No existe un modelo perfecto. Cada sistema debe equilibrar consistencia, rendimiento y disponibilidad.**

**Los modelos de consistencia son fundamentales en las CPUs modernas, sistemas operativos, bases de datos distribuidas y en los servicios de la nube.**

**Gracias por su  
atención!**

