



Dpto. Ciencias e Ingeniería de la Computación  
 Universidad Nacional del Sur

## ELEMENTOS DE BASES DE DATOS

Segundo Cuatrimestre 2014

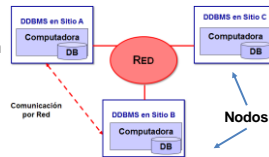
### Clase 21: Sistemas de Manejo de Bases de Datos Distribuidos (Parte I)



Mg. María Mercedes Vitturini  
 [mvitturi@uns.edu.ar]

## Bases de datos distribuidas

- Un **sistema de bases de datos distribuido (DDBMS ó SMBDD)** se compone de varios **sitios o nodos** levemente acoplados que no comparten componentes físicos.
- Los nodos se conectan a través de la red.
- Los nodos pueden variar en tamaño y función.
- Generalmente se ubican geográficamente separados.
- Cada DBMS de un sitio funciona independientemente de los otros.



Nodos

Comunicación por Red

RED

DDBMS en Sitio A  
Computadora DB

DDBMS en Sitio B  
Computadora DB

DDBMS en Sitio C  
Computadora DB

EBD2014\_21 - Mg. Mercedes Vitturini

## Bases de Datos Distribuidas

- **Bases de datos distribuidas homogéneas (SMBDD):**
  - El mismo software/esquema de base de datos en los sitios.
  - Los datos (instancias) están particionados y/o replicados en los sitios.
  - **Objetivo:** proveer los usuarios la vista de una sola base de datos ocultando detalles de distribución.
- **Bases de datos distribuidas heterogéneas:**
  - Diferentes software/esquema en los sitios.
  - **Objetivo:** integrar distintas bases de datos para proveer una función útil.

EBD2014\_21 - Mg. Mercedes Vitturini

## SMBDD: gestión de datos

**Replicación** – el sistema mantiene varias copias (réplicas) idénticas de una relación  $r$ . Cada réplica se almacena en un sitio diferente. Mejoras

- Tiempo de respuesta y tolerancia a fallos.

**Fragmentación** – una relación  $r$  se divide en varios fragmentos almacenados en sitios diferentes.

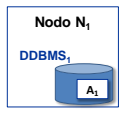
**Replicación y Fragmentación** – una relación  $r$  es particionada en varios fragmentos y el sistema mantiene varias copias idénticas de éstos fragmentos.

EBD2014\_21 - Mg. Mercedes Vitturini

## Datos Lógicos y Datos Físicos

- En DDBMS ó SMBDD se identifican los conceptos “*dato*”

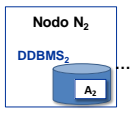
**DATO LÓGICO**



Nodo  $N_1$   
DDBMS<sub>1</sub>  
 $A_1$

A es el dato lógico

**DATO FÍSICO**



Nodo  $N_2$   
DDBMS<sub>2</sub>  
 $A_2$

$A_i$  es el dato físico

- Si de un dato existe una única copia  $\Rightarrow$  dato lógico  $\equiv$  dato físico.
- En entornos distribuidos homogéneos con replicación **la gestión de bloqueos debe realizarse a nivel de dato lógico.**
- El criterio de compatibilidad entre bloqueos es la misma.

EBD2014\_21 - Mg. Mercedes Vitturini

## Gestor de Bloqueos Distribuido

- Entre los cambios a incorporar en el SMDDBD está **cómo debe operar el Gestor de Bloqueos del SMDDBD si cuando existen múltiples copias de los datos**, de forma tal que:
  - **Múltiples transacciones** puedan obtener un **lock compartido sobre A**, pero **una única transacción** por vez puede obtener un **lock exclusivo sobre A**.
  - Además, en caso de actualización, asegurar hay que **todas las copias de un ítem de datos** se pueden **actualizar** y así permanecer iguales.

EBD2014\_21 - Mg. Mercedes Vitturini

## Gestión de Bloques en DDBMS

- Algunas alternativas para **implementar la traducción entre “lock de dato lógico” en “locks de datos físicos”** en entornos distribuidos con réplicas:
  - **Utilizar un Gestor de Bloqueos Centralizado.**
    - **Nodo Central**
  - **Utilizar Gestión de Bloqueos Distribuida.**
    - ROWA
    - Mayoría
    - K de n
  - **Otras propuestas híbridas**
    - Sitio primario (o copia primaria)
    - Token primario

EBD2014\_21 - Mg. Mercedes Vitturini

## Gestión de Bloques en DDBMS

- La comunicación entre los sitios o nodos se realiza por medio de **mensajes**.
- La cantidad de mensajes a transmitir “condiciona” la performance general del sistema. Se distinguen:
  - **Mensajes de Control:** que indican requerimientos o concesión de bloqueos, estados de cometido o de aborto, estados de deadlock, etc.
  - **Mensajes de Datos:** que contienen datos puros, leídos o de a ser escritos en la base de datos.
- Bajo ciertas condiciones los mensajes de control cuestan lo mismo que los de datos, pero en general los mensajes de datos son más caros.

EBD2014\_21 - Mg. Mercedes Vitturini

## Gestor de Bloqueos Centralizado: Nodo Central

- El DDBMS define como **gestor de bloqueos a un sitio S**.
- Si una transacción en un sitio  $S_i$  necesita bloquear un ítem de dato Q, envía el requerimiento a S, que determina si se puede conceder:
  - Si es compatible, S envía el mensaje al sitio  $S_i$  que lo solicitó.
  - Sino, el  $S_i$  quedará demorado hasta que se le pueda asignar el bloqueo.

### Ventajas ☺ :

- Implementación simple y control de deadlock distribuido simple.

### Desventajas ☹ :

- S es el “lock manager” se transforma en cuello de botella y el DDBMS es vulnerable al fallo del sitio S.

EBD2014\_21 - Mg. Mercedes Vitturini

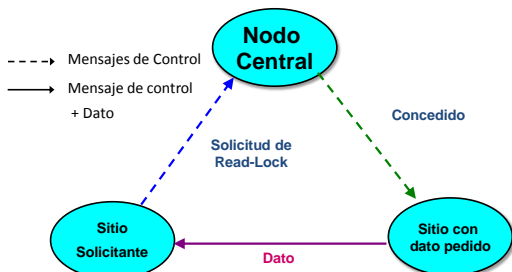
## Mensajes con Nodo Central

Para obtener un **read-lock**:

- $S_i$  envía un mensaje requiriendo un **read-lock** al nodo central S.
- Si el **bloqueo no es concedido**,  $S_i$  se queda esperando.
- Si el **bloqueo es concedido**, el nodo central S envía un mensaje a un sitio  $S_k$  que tiene una copia del ítem de dato.
- Luego, el sitio  $S_k$  envía un **mensaje con el valor del dato** al sitio que hizo el requerimiento.

EBD2014\_21 - Mg. Mercedes Vitturini

## Lock-S con Nodo Central



EBD2014\_21 - Mg. Mercedes Vitturini

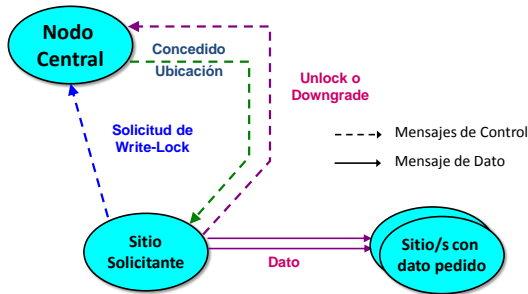
## Método de Nodo Central

Para obtener un **write-lock**:

- El sitio  $S_i$  que necesita actualizar un dato envía un mensaje requiriendo un **write-lock** al nodo central.
- Si **el bloqueo no es concedido**, el sitio  $S_i$  se queda esperando.
- Si **el bloqueo es concedido**, el nodo central le contesta con un **mensaje notificando** el/los sitio/s que tiene/n copia del ítem de dato. Luego, el/los sitio/s con la copia del ítem de dato reciben el nuevo valor a escribir.
- $S_i$  envía un **mensaje adicional liberando el bloqueo** de escritura una vez que el dato fue actualizado.

EBD2014\_21 - Mg. Mercedes Vitturini

### Lock-X con Nodo Central



EBD2014\_21 - Mg. Mercedes Vitturini

### Método de Nodo Central

#### Análisis del método Nodo Central

- ☹ Se requieren de mensajes de control extra: al nodo central y al nodo donde reside la copia del dato.
- ☹ El nodo central es el cuello de botella de la performance de la red, la mayoría de los mensajes vienen de él o van hacia él.
- ☹ La falla del sitio que opera de nodo central torna inoperativo al sistema.
- ☺ Es sencillo de implementar y de detectar y manejar situaciones de *deadlock*.

EBD2014\_21 - Mg. Mercedes Vitturini

### Gestión de Bloqueos Híbrida: Sitio primario

- Define un sitio primario o "nodo copia primaria" para cada ítem de dato.
- Así, la responsabilidad en la administración de los bloqueos para un ítem lógico A se deja a cargo de un sitio particular, sin importar cuántas copias del mismo dato existan.
  - Ejemplo: sea una base de datos de un banco y los nodos de la red sucursales del banco, es natural considerar el sitio primario del ítem cuenta a la sucursal que creó y tiene esa cuenta.
- Si el sitio primario S del ítem A no es el nodo S<sub>i</sub> donde se ejecuta la transacción, entonces S<sub>i</sub> envía un requerimiento de bloqueo al gestor de S y se espera por su respuesta.

EBD2014\_21 - Mg. Mercedes Vitturini

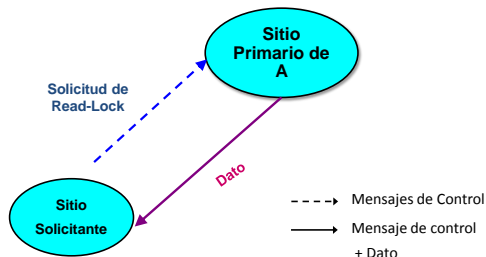
### Sitio primario

Varios sitios comparten la responsabilidad de administrar bloqueos:

- La gestión de bloqueos distribuida se implementa colaborando entre varios gestores de bloqueos ubicados en distintos nodos.
- Cada gestor de bloqueos controla el acceso a datos de los que tiene una copia local.
- Se define un protocolo para actualizar todas las copias.
- ☺ Ventajas:
  - Distribuye el trabajo,
  - El sistema es más robusto ante fallos.

EBD2014\_21 - Mg. Mercedes Vitturini

### Lock-S con Sitio Primario



EBD2014\_21 - Mg. Mercedes Vitturini

### Gestión de Bloques en DDBMS

- Algunas alternativas para implementar la traducción entre "lock de dato lógico" en "locks de datos físicos" en entornos distribuidos con réplicas:
  - Utilizar un Gestor de Bloqueos Centralizado.
    - Nodo Central ✓
  - Utilizar Gestión de Bloqueos Distribuida.
    - ROWA
    - Mayoría
    - K de n
  - Otras propuestas híbridas
    - Sitio primario (o copia primaria) ✓
    - Token primario

EBD2014\_21 - Mg. Mercedes Vitturini

## Gestión de bloqueos distribuida: Read-Locks-One / Write-Locks-All

- Para obtener un **read-lock** (*lock-s*) sobre un ítem de dato A, la transacción debe **obtener un lock-S sobre una de las copias de A** (dato físico).
- Para obtener un **write-lock** (*lock-x*) sobre un ítem de dato A, la transacción debe obtener el **lock sobre todas las copias** de A (dato físico).

Las reglas para conceder bloqueos son:

- Un **read-lock** se consigue sobre una copia si ninguna otra transacción tiene un write-lock sobre la copia.
- Un **write-lock** es concedido sobre una copia si ninguna otra transacción tiene un read-lock o un write-lock sobre la copia.

EBD2014\_21 - Mg. Mercedes Vitturini

## ROWA - Análisis

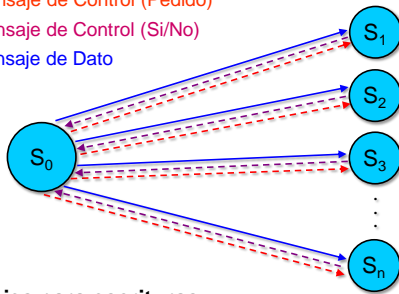
Si existen  **$n$  copias de un dato A**.

- Para obtener **read-lock sobre A**, se necesita conocer un sitio donde exista una copia de A para enviarle el requerimiento de bloqueo. Si se concede el bloqueo: **1 mensaje de control** y **1 mensaje de datos**. La concesión del bloqueo viene con el dato.
- Para obtener **write-lock sobre A**, se necesitan enviar  **$2n$  mensajes de control** ( $n$  para requerir el bloqueo y  $n$  de concesión del mismo) y  **$n$  mensajes de datos**.

EBD2014\_21 - Mg. Mercedes Vitturini

### ROWA: lock-X

- > Mensaje de Control (Pedido)
- > Mensaje de Control (Si/No)
- > Mensaje de Dato

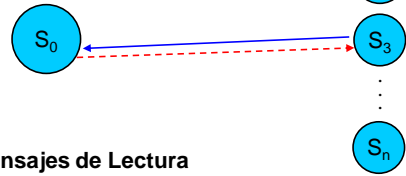


Mensajes para escrituras

EBD2014\_21 - Mg. Mercedes Vitturini

### ROWA: lock-s

- > Mensaje de Control (Pedido)
- > Mensaje de Control (Si) + Dato



Mensajes de Lectura

EBD2014\_21 - Mg. Mercedes Vitturini

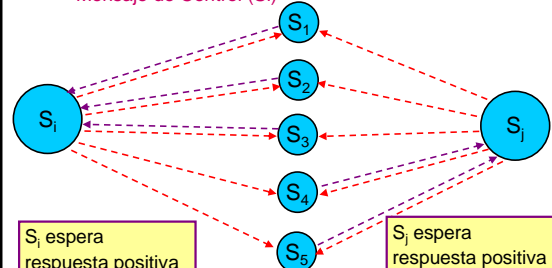
## ROWA

- ROWA se comporta apropiadamente en entornos donde **predominan las lecturas**.
  - **Ejemplo:** una guía telefónica on-line, donde los datos se actualizan en períodos fijos y las lecturas son en todo momento las operaciones predominantes.
- Si se solicita más de un pedido de escritura (lock-X) simultáneamente y el mismo no puede ser otorgado, si se decide esperar se puede entrar en **deadlock**.
- Cuando se solicita un pedido de lectura y el mismo no puede ser otorgado, en general se **espera** (lo más probable es que otro sitio esté escribiendo el dato).

EBD2014\_21 - Mg. Mercedes Vitturini

## Deadlocks en escrituras en ROWA

- > Mensaje de Control (Pedido)
- > Mensaje de Control (Si)



EBD2014\_21 - Mg. Mercedes Vitturini

## Majority Locking

- Para obtener un **read-lock sobre A**, una transacción debe obtener un read-lock sobre **la mayoría de las copias de A**.
- Para obtener un **write-lock sobre A**, una transacción debe obtener un write-lock sobre **la mayoría de las copias de A**.

Las reglas para conceder bloqueos son iguales al protocolo anterior. Si se asume que el número ( $n$ ) de copias la mayoría es el techo de  $(n+1)/2$ .

**Ejemplo:** la mayoría para  $n = 5$  es 3, y la mayoría para  $n = 6$  es 4.

EBD2014\_21 - Mg. Mercedes Vitturini

## Majority Locking - Análisis

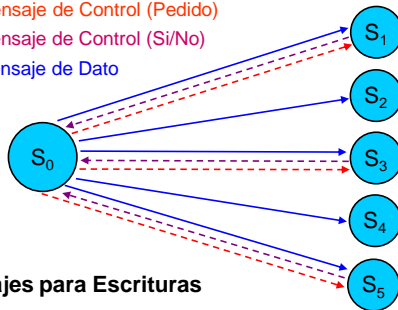
Si existen  $n$  copias de un dato A.

- Para realizar un **write-lock A**, se envían  $n+1$  mensajes de control ( $(n+1)/2$  para requerir el bloqueo y  $(n+1)/2$  de concesión del mismo) y  $n$  mensajes de datos (se escriben todas las copias).
- El número de mensajes de control es similar si se realiza un **read-lock** pero se atiende sólo **1 mensaje de datos**.
- Si la transacción corre en el sitio de una de las copias, se pueden omitir algunos mensajes (no se transfieren por la red).

EBD2014\_21 - Mg. Mercedes Vitturini

## Majority Locking: lock-x

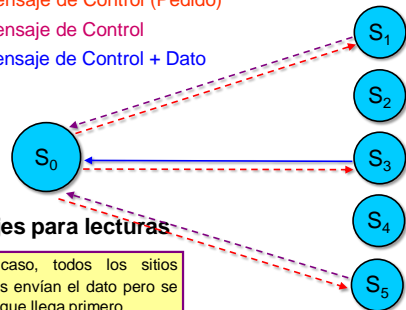
- Mensaje de Control (Pedido)
- Mensaje de Control (Si/No)
- Mensaje de Dato



EBD2014\_21 - Mg. Mercedes Vitturini

## Majority Locking: lock-s

- Mensaje de Control (Pedido)
- Mensaje de Control
- Mensaje de Control + Dato



EBD2014\_21 - Mg. Mercedes Vitturini

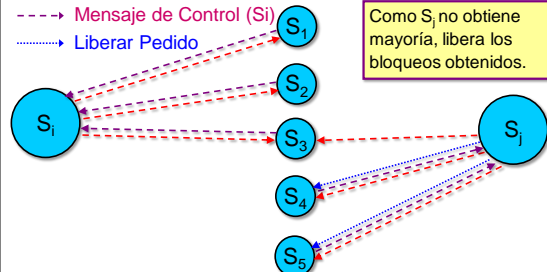
## Majority Locking

- Este protocolo se comporta de manera apropiada si **predominan las escrituras**.  
 –Ejemplo: un sistema de reserva de pasajes, donde los datos se modifican constantemente.
- Cuando se solicitan simultáneamente pedidos, sólo un sitio puede obtener la mayoría por lo que **existe menor riesgo de deadlock**.
- Si se solicita un dato y fracasa el pedido, se puede esperar que se liberen copias para obtener la mayoría, o bien se aborta la transacción.
- En caso de espera, debe garantizarse que un sitio no entre en estado de inanición (en espera continua).

EBD2014\_21 - Mg. Mercedes Vitturini

## Menor riesgo de deadlocks en escrituras

- Mensaje de Control (Pedido)
- Mensaje de Control (Si)
- ..... Liberar Pedido



EBD2014\_21 - Mg. Mercedes Vitturini

### Estrategia $k$ -de- $n$

Si existen  $n$  copias del dato. Sea  $k$  un valor tal que  $n/2 < k \leq n$ .

- Para obtener un **write-lock A**, una transacción debe obtener un write-lock sobre  $k$  copias de  $A$ .
  - Para obtener un **read-lock A**, una transacción debe obtener un read-lock sobre un total de  $n-k+1$  copias de  $A$ .
- ✓ No es posible que existan read-locks y write-locks simultáneamente (se necesitarían  $n+1$  copias de  $A$ ).  
 ✓ Tampoco pueden existir dos write-locks simultáneamente ( $k > n/2$ ).

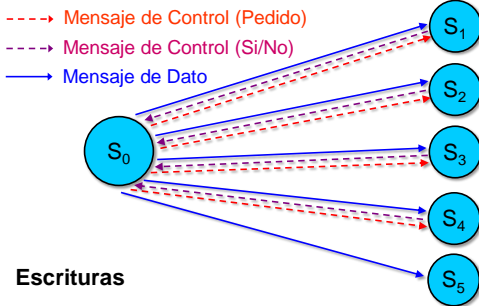
EBD2014\_21 - Mg. Mercedes Vitturini

### Estrategia $k$ -de- $n$

- La estrategia  $n$ -de- $n$  deriva en **Write-Locks-All**.
- La estrategia  $(n+1)/2$ -de- $n$  deriva en **Majority Locking**.
- A medida que  $k$  se incrementa, el protocolo se desempeña mejor en situaciones en donde se realizan lecturas más frecuentemente.
- A medida que  $k$  se decrementa, el protocolo se desempeña mejor en situaciones donde predominan las escrituras.

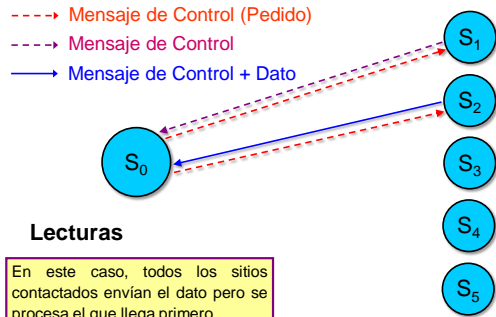
EBD2014\_21 - Mg. Mercedes Vitturini

### Estrategia 4-de-5: lock-x



EBD2014\_21 - Mg. Mercedes Vitturini

### Estrategia 4-de-5: lock-s



EBD2014\_21 - Mg. Mercedes Vitturini

### Gestión de Bloques en DDBMS

- Algunas alternativas para implementar la traducción entre "lock de dato lógico" en "locks de datos físicos" en entornos distribuidos con réplicas:
  - Utilizar un Gestor de Bloqueos Centralizado. ✓
    - Nodo Central
  - Utilizar Gestión de Bloqueos Distribuida. ✓
    - ROWA
    - Mayoría
    - K de n
  - Otras propuestas híbridas
    - Sitio primario (o copia primaria) ✓
    - Token primario

EBD2014\_21 - Mg. Mercedes Vitturini

### Tokens de Copia Primaria

- Este método asume la existencia de **read-tokens** y **write-tokens**, o privilegios que los nodos de la red pueden obtener, en beneficio de sus transacciones y con el fin de acceder a los ítems.
- Para un ítem  $A$ , puede existir **sólo un write-token**. Si no existe un write-token, puede existir **cualquier número de read-tokens**.
  - Si un sitio tiene un **write-token** para  $A$ , entonces puede conceder un **read-lock** o un **write-lock** para  $A$ .
  - Si un sitio tiene un **read-token** para  $A$ , entonces solamente puede conceder un **read-lock** para  $A$ .

EBD2014\_21 - Mg. Mercedes Vitturini

### Tokens de Copia Primaria

- Si una transacción en un sitio  $N$  desea un *write-lock* para  $A$ , debe obtenerlo para el sitio.
  - Si el *write-token* está en ese sitio entonces no hace nada.
  - Si el *write-token* no está en ese sitio entonces

Continua ...

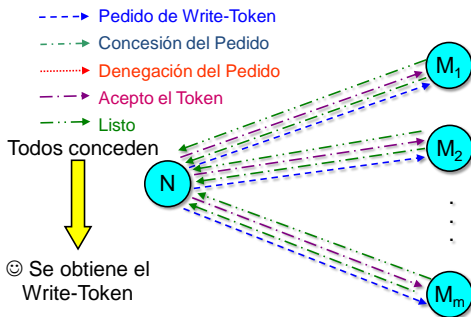
EBD2014\_21 - Mg. Mercedes Vitturini

### El *write-token* no está en el sitio entonces ...

- $N$  envía un mensaje a todos los sitios requiriendo el *write-token*.
- Cada sitio  $M$  que recibe el requerimiento y contesta:
  - (a)  $M$  no tiene un *read/write-token* para  $A$  o está por liberarlo de modo que  $N$  puede obtener el *write-token*.
  - (b)  $M$  tiene un *read/write-token* para  $A$  y no lo liberará (otra transacción lo está usando o ha sido reservado a otro sitio).
- Si todos los sitios contestan (a),  $N$  puede obtener el *write-token* y envía un mensaje a cada sitio diciendo que aceptó el *write-token* y que deberían destruir el que ellos tienen.
- Si algunos contestan (b),  $N$  no puede obtener el *write-token* y debe enviar un mensaje a los sitios que contestaron (a) para que cancelen la reserva sobre  $A$ .

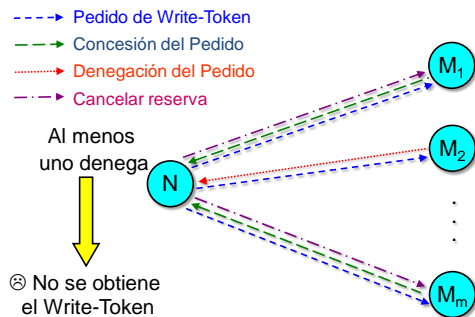
EBD2014\_21 - Mg. Mercedes Vitturini

### Token de Copia Primaria



EBD2014\_21 - Mg. Mercedes Vitturini

### Token de Copia Primaria



EBD2014\_21 - Mg. Mercedes Vitturini

### Comparación de métodos

MÉTODO	MENSAJES DE CONTROL EN ESCRITURA	MENSAJES DE CONTROL EN LECTURAS	OBSERVACIONES
ROWA	$2n$	1	Bueno en ambientes donde predominan lecturas
Mayoría	$\geq n+1$	$\geq n$	Compensa mensajes en lecturas y escrituras
Nodo Central	3	2	Simple y vulnerable a fallos
Sitio primario	2	1	Eficiente y vulnerable a fallos
Token primario	0-4m	0-4m	Se adapta a cambios temporarios

$n$ : número de copias del dato       $m$ : número de nodos en la red

EBD2014\_21 - Mg. Mercedes Vitturini

### Temas de la clase de hoy

- Sistemas Distribuidos
  - Protocolos de bloqueo con replicación
- Bibliografía
  - “Principles of Database and Knowledge-Base Systems” – J. Ullman. Capítulo 10.
  - “Database Systems Concepts” – A. Silberschatz. Capítulo 19 (ed. 2010)

EBD2014\_21 - Mg. Mercedes Vitturini