

Sistemas de información Bases de datos distribuidas

Grado en Ingeniería en Informática Programa Conjunto MAT-INF



Curso 2023-2024

Fernando Tricas García (ftricas@unizar.es)
Raquel Trillo Lado (raqueltl@unizar.es)
Carlos Tellería Orriols (telleria@unizar.es)
Dpto. Informática e Ingeniería de Sistemas



Bases de Datos Distribuidas, Interoperantes y Federadas



Guion

- Introducción
- Bases de datos distribuidas
 - Ventajas y desventajas
 - > Diseño de BBDD: Horizontales, Verticales e Híbridas
 - > Arquitectura
- Bases de datos federadas
- Bases de datos interoperantes
- > Referencias



BASE DE DATOS CENTRALIZADA



CLIENTES
PRODUCTOS
COMPRAS
GASTOS

¿PROBLEMAS?

- Cuellos de botella
- Latencias (¿varias sedes?)
- Distintas frecuencias de acceso
- ..



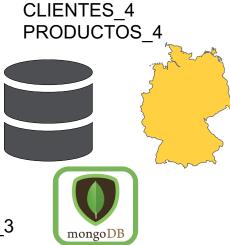
Introducción

BASE DE DATOS DISTRIBUIDA





CLIENTES_3 PRODUCTOS_3 COMPRAS_3 GASTOS 3







CLIENTES_2







CLIENTES_2 PRODUCTOS_2 COMPRAS_2 GASTOS 2





Introducción

- > Tecnología de BBDD (tradicional)
 - Centralización de datos
 - ➤ Varios ficheros → Una base de datos



Evitar inconsistencias

- > Redes de computadores
 - Distribución/compartición de recursos (ej. datos)
 - ➤ BD centralizada → BD de enfoque distribuido

Mantener integridad

- > BD distribuidas, interoperantes y federadas: unión de estas dos aproximaciones
 - La tecnología de BBDD busca la INTEGRACIÓN de los datos y no la CENTRALIZACIÓN



Guion

- > Introducción
- Bases de datos distribuidas
 - Ventajas y desventajas
 - > Diseño de BBDD: Horizontales, Verticales e Híbridas
 - > Arquitectura
- Bases de datos federadas
- Bases de datos interoperantes
- > Referencias



Un sistema de **BBDD** distribuidas es una colección de varias BBDD que se encuentran lógicamente interrelacionadas y desplegadas sobre una red de ordenadores.

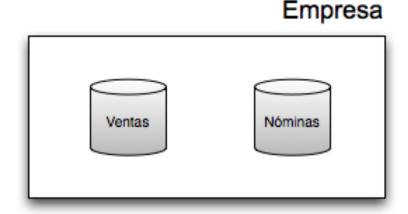
➤ Un sistema de gestión de bases de datos distribuidas (SGBDD) es el software que permite el manejo de sistemas de BBDD distribuidas (BDD) y que hace dicha distribución **transparente** al usuario (semántica vs implementación)



- > NO son Sistemas de BD distribuidas:
 - ➤ Un sistema de multiprocesado (BD paralelas)
 - Un sistema de BD que reside en uno de los nodos de una red. Eso es una BD centralizada accesible a través de la red
 - ➤ Un conjunto de BD que pueden comunicarse unas con otras donde no existe un esquema global



> ¿Es la siguiente una BD distribuida?



No, si no existe un mecanismo para ver ambas como una única base de datos



Ventajas



> Transparencia de red

- > El usuario no debe ser consciente del uso de la red
- > Transparencia de localización: dónde están los datos y qué lenguajes "locales" son necesarios
- > Transparencia de nombres: nombres únicos (espacio de nombres) en todo el sistema distribuído, independientes de la localización

> Transparencia de fragmentación

➤ El usuario no debe ser consciente de la existencia de varios depósitos de datos

> Transparencia de replicación

El usuario no debe ser consciente de la existencia de varias copias de datos



Ventajas (cont.)



- La distribución puede ser la organización más natural
- Mayor **fiabilidad** y **disponibilidad** (puede haber replicación, no se depende de un único nodo)
- > Autonomía local: es posible establecer políticas locales de acceso a los datos
- Más eficiencia al acceder a los datos locales, frente a una aproximación centralizada



Ventajas (cont.)



- Economía: mejor varias máquinas en red que un *mainframe*
- ➤ **Escalabilidad**: más posibilidades de expansión → añadir más recursos a la red
- Compartición y disponibilidad de datos (por encontrarse en una red)



Desventajas



- Aún falta de experiencia en el diseño de BD distribuidas
- Complejidad
 - Todos los problemas de las BD centralizadas +
 - Seguridad en redes
 - > Transacciones distribuidas
 - Control distribuido
- Coste: HW + SW + comunicaciones
- > Dificultad de cambio
 - Pequeñas/medianas empresas ya tienen BD centralizadas



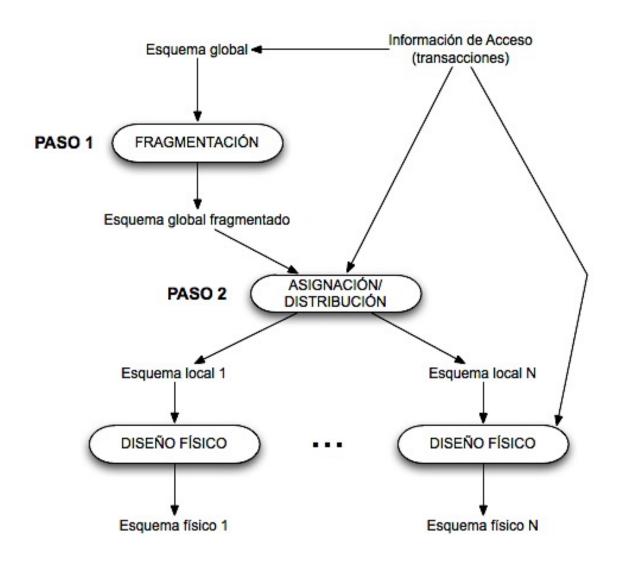
- Diseño top-down
 - Diseño de un esquema global (E/R, Relacional)
 - > Fragmentación del esquema global
 - Elaboración de los esquemas locales y **asignación** de los fragmentos a los esquemas locales
- > Fragmento: es la unidad a distribuir
 - > Puede ser:
 - Parte de una tabla
 - Una tabla entera
 - Un conjunto de tablas

Incrementa el nivel de concurrencia de las transacciones



Algunas transacciones se degradarán si tienen que trabajar con varios fragmentos

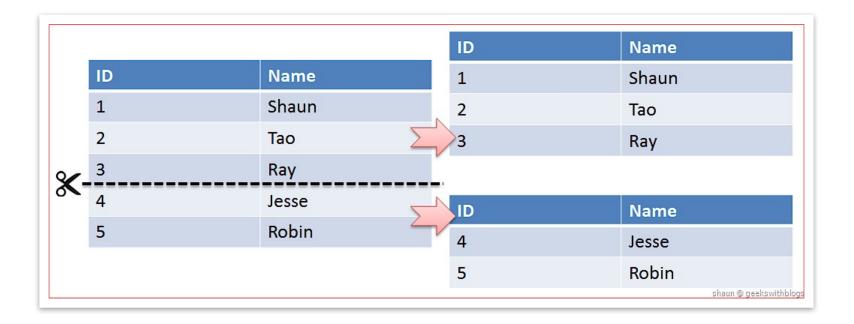






Tipos de Fragmentación

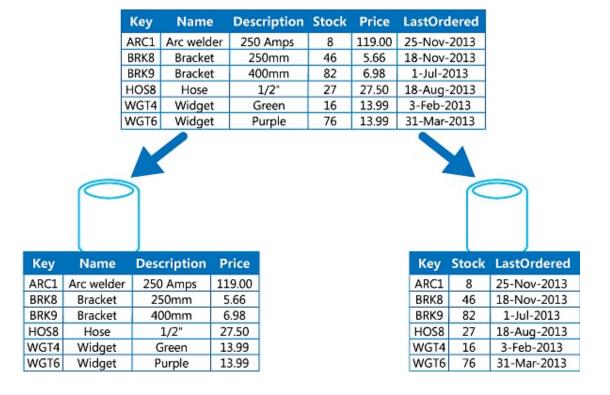
> Horizontal: basada en encontrar condiciones de selección





Tipos de Fragmentación

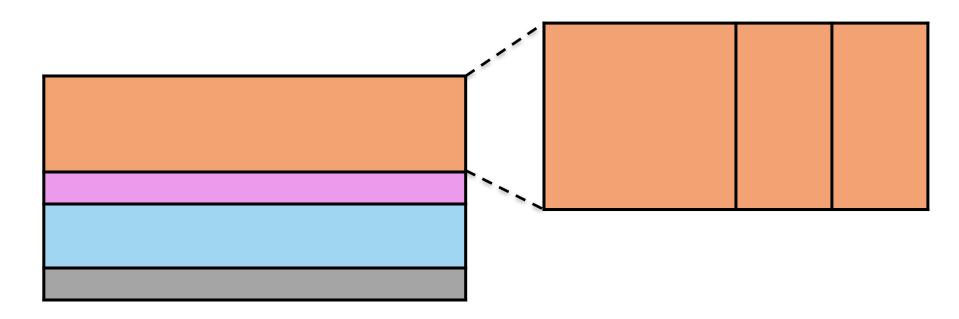
Vertical: basada en encontrar conjuntos de atributos a proyectar





Tipos de Fragmentación

> Híbrida: primero horizontal y después vertical



Horizontal Vertical



Fragmentación debe ser...

> Completa

Todo elemento de la relación debe estar en alguno de los fragmentos

Reconstruíble

La relación inicial debe poder reconstruirse aplicando operadores sobre los fragmentos (p.ej. con operaciones JOIN, UNION...)

> Con intersección vacía (disjointness)

La intersección de los fragmentos debe ser vacía (a excepción de las claves)



Asignación

- > Asignar fragmentos a los esquemas locales
 - > Sin replicación: todo fragmento reside en un único nodo
 - Positivo para actualizaciones
 - Negativo para consultas
 - > Replicación total: todos los fragmentos residen en todos los nodos
 - Positivo para consultas
 - Negativo para actualizaciones
 - > Replicación parcial
 - Compromiso entre actualizaciones y consultas



Asignación

	REPLICACIÓN COMPLETA	REPLICACIÓN PARCIAL	SIN REPLICACIÓN
PROCESAMIENTO DE CONSULTAS	Más fácil	Más difícil	Más difícil
CONTROL DE CONCURRENCIA	Difícil	Más difícil	Más fácil
DISPONIBILIDAD DE LOS DATOS	Muy alta	Alta	Baja



Arquitectura: Factores (I)

1. Distribución

- Una BD es distribuida si está dividida en distintos componentes (integrados)
- BD distribuida != varias BD no integradas
- Los componentes distribuidos que constituyen una BD distribuida son, a su vez, bases de datos (BD componentes o locales)
- Las BD componentes tendrán un grado de autonomía local determinado



Arquitectura: Factores (II)

- 2. Autonomía: tipo de control que el SGBD tiene sobre cada BD local
 - Autonomía de diseño: existe si los administradores de la BD pueden cambiar el esquema conceptual de sus BD independientemente de si forman parte de un sistema distribuido o no
 - Autonomía de comunicación: si se puede decidir localmente cuándo comunicarse con los otros SGBD locales
 - Autonomía de ejecución: si se puede ejecutar transacciones globales y locales en el orden que se quiera
 - Autonomía de participación: si se puede decidir cómo participar en el sistema distribuido



Arquitectura: Factores (III)

3. Heterogeneidad

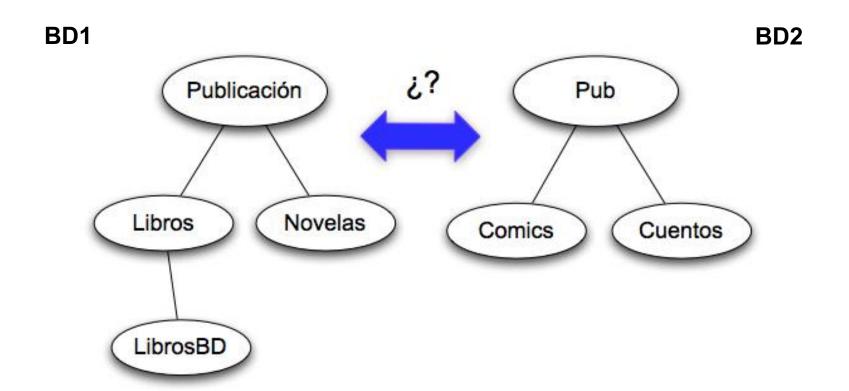
- > Distinto hardware, SO, software de comunicaciones
- > Distinto modelo de datos (relacional, jerárquico, en red, OO, ...)
- Distintos SGBD (aunque sean del mismo modelo)



Guion

- > Introducción
- Bases de datos distribuidas
 - Ventajas y desventajas
 - > Diseño de BBDD: Horizontales, Verticales e Híbridas
 - > Arquitectura
- Bases de datos federadas
- Bases de datos interoperantes
- > Referencias







BD1:

Coches(numSerie, modelo, color)

CochesConCambioAutom(numSerie, modelo, color)

BD2:

Coches(numSerie, modelo, color, tieneCambioAutom)



- > Formadas por BD autónomas
- Proporcionan un esquema global
- > El esquema global se obtiene de abajo a arriba
 - Los esquemas locales son pre-existentes y se integran en un esquema global
- > No hay que fragmentar, y la redundancia probablemente ya existe



Arquitectura: Factores (III)

3. Heterogeneidad

- > Distinto hardware, SO, software de comunicaciones
- Distinto modelo de datos (relacional, jerárquico, en red, OO, ...)
- Distintos SGBD (aunque sean del mismo modelo)
- Heterogeneidad semántica
 - Sinonimia: elementos iguales con distintos nombres
 - Homonimia: elementos distintos con el mismo nombre
 - > Otras relaciones semánticas (hiperonimia, hiponimia, agregación, etc.)
 - ➤ El mismo elemento del mundo real puede ser representado como entidad o atributo, atributos con tipos diferentes, etc.
 - Puede existir tanto a nivel intensional como extensional



Arquitectura: Factores (IV)

3. Heterogeneidad (ejemplos)

- Sinónimos a nivel intensional (Reglas)
 - La tabla "alumnos" en BD1 representa la misma abstracción que "estudiantes" en BD2
 - > El atributo "alumnos.id" es equivalente al atributo "estudiantes.dni"
- Sinónimos a nivel extensional (entre objetos de distintas BD) (Hechos)
 - ➤ El valor "J. Pérez" de "alumnos.nombre" es equivalente al valor "Pérez, J." de "estudiantes.nombre"
 - A veces es imposible (ej. Nombre, "Javier Pérez" y "J. Pérez")
 - O es difícil de implementar (ej. DNI y nº de SS)



- El problema de obtener un esquema global a partir de N esquemas locales se divide en dos:
 - > Traducción: cada esquema local se traduce a un modelo canónico
 - > Integración: los esquemas locales se integran en uno solo
- Este es un tema de investigación a día de hoy



- El modelo de datos (canónico) utilizado para expresar el esquema global es muy importante:
 - ➤ No hay que olvidar que las bases de datos locales pueden ser heterogéneas (distintos modelos de datos)
 - > Se utilizan modelos más ricos semánticamente que el relacional: OO, modelos funcionales, semánticos, etc.

➤ Supongamos que los esquemas locales son relacionales y se usa como modelo canónico el modelo E/R Extendido de Chen:

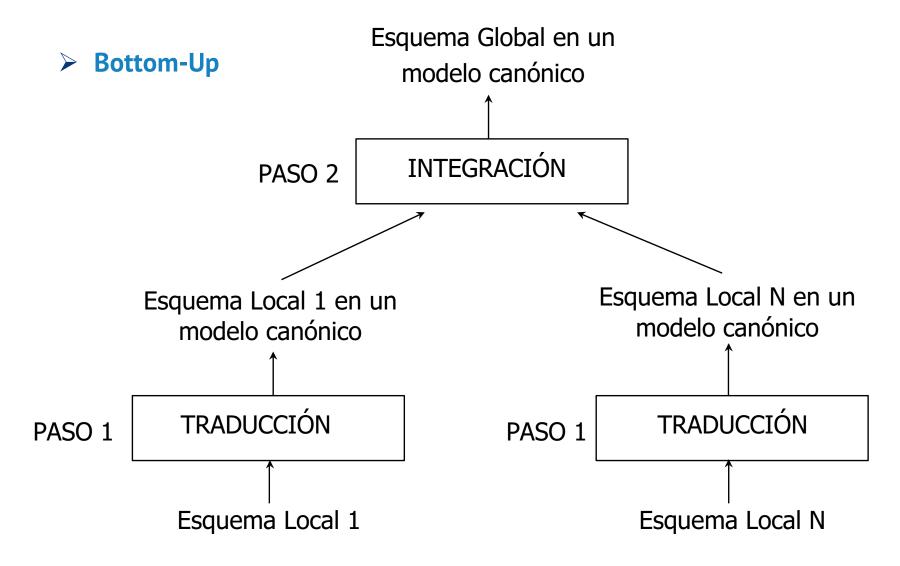
> Traducción

- A partir de tablas y atributos relacionales (esquema exportado) se identifican entidades, relaciones y atributos (enriquecimiento semántico)
- Pueden aparecer nuevas entidades (especializaciones / generalizaciones, etc.)

> Integración

Aplicación de las propiedades semánticas entre las entidades y relaciones de distintos esquemas locales canónicos (sinonimia, unión, generalización/especialización...)







- Procesamiento de consultas
 - Las consultas realizadas sobre el esquema global deben responderse sobre los esquemas locales
- Información de enlace
 - Relación entre los elementos del esquema global y los de los esquemas locales
 - Necesaria para poder responder a las consultas y construida en las fases de traducción e integración



Esquemas locales

BD1.estudiantes(nombre, dni, curso)

BD2.alumnos(id, nombre, tutor)

Esquemas canónicos en E/R Extendido

estudiantes (Inf. enlace: BD1.estudiantes)

alumnos (Inf. enlace: BD2.alumnos)

est-Segundociclo (Inf. enlace: SELECT *

FROM BD1.estudiantes

WHERE curso>3)

Esquema global o integrado ("estudiantes" sinónimo de "alumnos")

Estudiantes (Inf. enlace: SELECT nombre, dni, curso, NULL FROM

BD1.estudiantes UNION

SELECT nombre, id, NULL, tutor FROM BD2.alumnos)

est-2ºciclo (Inf. enlace: SELECT * FROM BD1.estudiantes WHERE curso>3)



Guion

- > Introducción
- Bases de datos distribuídas
 - Ventajas y desventajas
 - > Diseño de BBDD: Horizontales, Verticales e Híbridas
 - > Arquitectura
- Bases de datos federadas
- Bases de datos interoperantes
- > Referencias



Bases de datos interoperantes

- > Formados por BD autónomas
- No proporcionan esquema global sino lenguajes de acceso a BD
- > El usuario es consciente de que trabaja con varias BD



Guion

- **■**Introducción
- Bases de datos distribuídas
 - ■Ventajas y desventajas
 - Diseño de BBDD: Horizontales, Verticales e Híbridas
 - Arquitectura
- Bases de datos federadas
- **■**Bases de datos interoperantes
- Referencias



Referencias

■R.A. Elmasri, SB. Navathe Fundamentos de Bases de Datos, 3ª edición, Addison- Wesley, 2000.



Sistemas de información

Grado en Ingeniería en Informática Programa Conjunto MAT-INF



Universidad Zaragoza



Curso 2023-2024

Fernando Tricas García (ftricas@unizar.es)

Raquel Trillo Lado (raqueltl@unizar.es)

Carlos Tellería Orriols (telleria@unizar.es)

Dpto. Informática e Ingeniería de Sistemas