

### **Arquitectura de los Sistemas de Bases de Datos Distribuidas.**

La arquitectura de un sistema define su estructura. Esto significa que se identifican los componentes del sistema, se especifica la función de cada componente y se definen las interrelaciones e interacciones entre estos componentes. La especificación de la arquitectura de un sistema requiere la identificación de los diversos módulos, con sus interfaces e interrelaciones, en términos del flujo de datos y control a través del sistema.

M. Tamer Ozsu y Patrick Vakduriez describen en su libro 3 arquitecturas de BDD

1. Sistemas cliente/servidor
2. BDD punto a punto
3. Sistemas multibases de datos

#### **Sistemas cliente/servidor:**

Los DBMS cliente/servidor ingresaron a la informática a principios de la década de 1990 y han tenido un impacto significativo tanto en la tecnología SMDDB como en la forma en que hacemos informática. La idea general es muy simple y elegante: distinguir la funcionalidad que debe proporcionarse y dividir estas funciones en dos clases: funciones de servidor y funciones del cliente. Esto proporciona una arquitectura de dos niveles que facilita gestionar la complejidad de los SMDDB modernos y la complejidad de la distribución.

Los servidores realizan el procesamiento y optimización de queries, manejo de transacciones y almacenamiento de los datos en cada nodo..

Los clientes manejan la interfaz al usuario, manejo de datos en caché y manejo de bloques de transacciones de la caché.

En el modelo cliente/servidor, un cliente interrumpirá la comunicación con su servidor habitual y conectará con otros servidores cuando desee acceder a otras bases de datos. En un entorno de bases de datos distribuidas, cuando se realiza una petición de información (transacción SQL) a un servidor determinado, éste involucrará a los servidores que sean necesarios para satisfacer la petición del cliente.

La capacidad para separar la lógica de la aplicación de la gestión de la base de datos y repartirlas en dos CPUs, permite a los sistemas cliente disponer de más potencia que a su vez les permitirá ejecutar los nuevos entornos gráficos, proporcionando al usuario un acceso más sencillo e intuitivo a la información que necesita. Al mismo tiempo, están disponibles toda una serie de nuevas prestaciones a nivel de gestores de bases de datos en cuanto a informática distribuida.

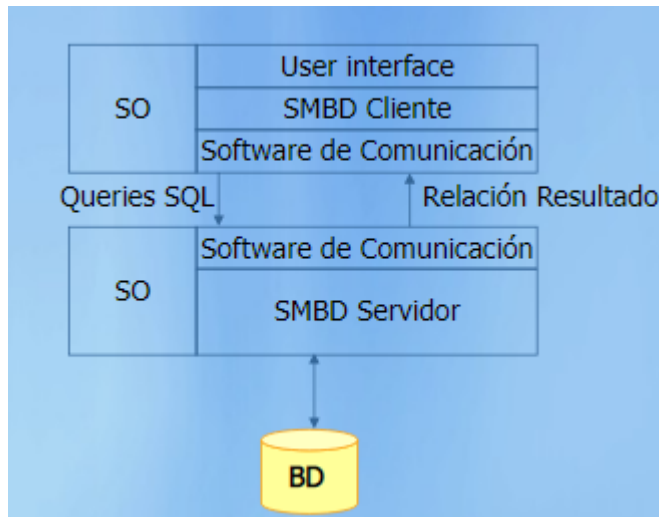
Las ventajas de la arquitectura cliente/servidor son tan numerosas, que la mayoría de las compañías no pueden ignorarlas a la hora de plantearse una evolución en sus sistemas de gestión de información.

Una ventaja importante del lenguaje SQL es su eficacia en entornos de red, lo que implica una mejora en el rendimiento del gestor de base de datos. Cuando se emplean servidores de archivos tradicionales, al estilo de las redes clásicas de PCs, una gran parte del archivo de datos viaja por la red desde el servidor de archivos a la máquina cliente. Al utilizar en el servidor un

gestor de bases de datos SQL, sólo los datos implicados en la consulta (query) -normalmente un fragmento de tabla o tablas- viajan a la máquina cliente.

Las dos arquitecturas cliente - servidor diferentes son:

- Servidor único Cliente múltiple
- Servidor Múltiple Cliente Múltiple



### Sistemas punto a punto:

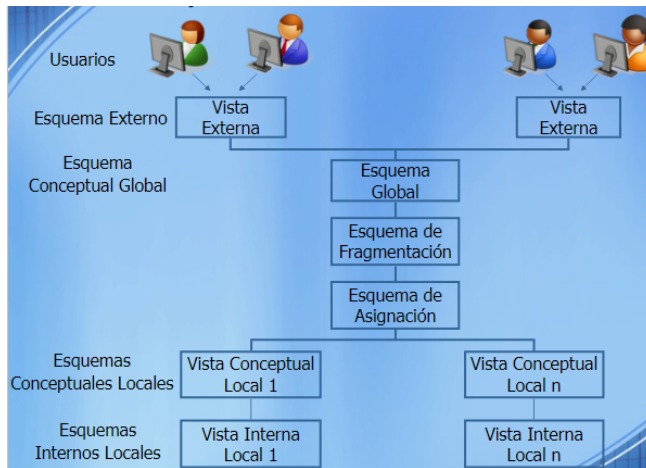
Después de una década de popularidad de la computación cliente/servidor, el modelo punto a punto ha hecho un regreso en los últimos años (impulsado principalmente por las aplicaciones para compartir archivos) y algunos incluso han posicionado la gestión de datos punto a punto como una alternativa a los DBMS distribuidos.

En estos sistemas, cada par actúa tanto como cliente como servidor para impartir servicios de base de datos. Los compañeros comparten sus recursos con otros compañeros y coordinan sus actividades. Esta arquitectura generalmente tiene cuatro niveles de esquemas.

- Esquema conceptual global: representa la vista lógica global de los datos.
- Esquema conceptual local: representa la organización lógica de datos en cada sitio.
- Esquema interno local: representa la organización de datos físicos en cada sitio.
- Esquema externo: representa la vista del usuario de los datos.

Si bien esto puede ser una exageración, los sistemas punto a punto modernos tienen dos diferencias importantes con sus parientes anteriores. El primero es la masiva distribución en los sistemas actuales. Mientras que en los primeros días nos enfocamos en unos pocos (quizás como máximo decenas de) sitios, los sistemas actuales consideran miles de sitios.

El segundo es la heterogeneidad inherente de cada aspecto de los sitios y su autonomía. Mientras esto siempre ha sido una preocupación de las bases de datos distribuidas, como se discutió anteriormente, junto con la distribución masiva, la heterogeneidad del sitio y la autonomía adquieren un significado adicional.



### Sistemas multibases de datos:

Los sistemas de multibases de datos (MDBS) representan el caso en el que los DBMS individuales (ya sean distribuidos o no) son totalmente autónomos y no tienen concepto de cooperación; que puede ni siquiera “saber” de la existencia del otro o cómo hablar entre ellos.

Este sistema se puede expresar a través de seis niveles de esquemas

- Nivel de vista de múltiples bases de datos: representa múltiples vistas de usuario que comprenden subconjuntos de la base de datos distribuida integrada.
- Nivel conceptual de base de datos múltiple: representa una base de datos múltiple integrada que se compone de definiciones de estructuras de bases de datos múltiples lógicas globales.
- Nivel interno de base de datos múltiple: representa la distribución de datos en diferentes sitios y la asignación de base de datos múltiple a datos locales.
- Nivel de vista de la base de datos local: representa la vista pública de los datos locales.
- Base de datos local Nivel conceptual: Representa la organización de datos locales en cada sitio.
- Nivel interno de la base de datos local: representa la organización física de los datos en cada sitio

Las diferencias en el nivel de autonomía entre los multi-DBMS distribuidos y los DBMS distribuidos también se reflejan en sus modelos arquitectónicos. La diferencia fundamental se relaciona con la definición del esquema conceptual global. En el caso de los DBMS distribuidos lógicamente integrados, el esquema conceptual global define la vista conceptual de toda la base de datos, mientras que en el caso de distribución multi-DBMS, representa solo la colección de algunas de las bases de datos locales que cada DBMS local quiere compartir.

El procesamiento de consultas en un sistema multibase de datos es la pieza más importante para la operación del sistema.

Tres pasos son necesarios para procesar una consulta global:

Una consulta global es descompuesta en subconsultas de manera que los datos necesarios por cada subconsulta estén disponibles desde cada SBDC (sistema de base de datos componentes).

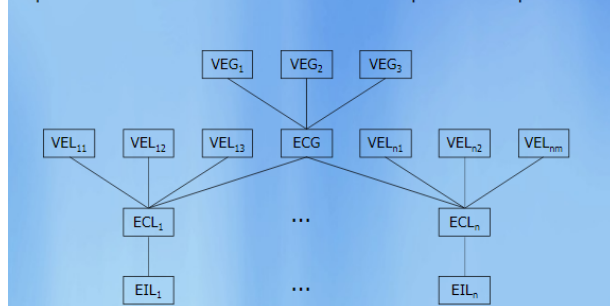
Después cada subconsulta es trasladada a una consulta o consultas de el SBDC y enviada o enviadas al SBDC.

Los resultados retornados por la subconsultas son combinados para dar respuesta a la consulta global.

La Heterogeneidad de las BD es inevitable cuando diferentes tipos de BD coexisten en una organización que trata de compartir datos entre éstas. BDD heterogéneamente:

El tratamiento de la información ubicada en bases de datos distribuidas heterogéneas exige una capa de software adicional por encima de los sistemas de bases de datos ya existentes. Esta capa de software se denomina sistema de bases de datos múltiples. Puede que los sistemas locales de bases de datos empleen modelos lógicos y lenguajes de definición y de tratamiento de datos diferentes, y que difieran en sus mecanismos de control de concurrencia y de administración de las transacciones.

Arquitectura de Multi Bases de Datos con un Esquema Conceptual Global



## Conclusión:

El tipo de arquitectura que se decida usar dependerá del tipo de aplicación que se quiera usar, la escalabilidad y de los recursos que se tengan.

Las bases de datos distribuidas son cada vez más usadas por las empresas y podría decirse que dan una ventaja competitiva frente a los sistemas centralizados, siempre y cuando la empresa tenga necesidad de usar una base de datos de este tipo. Lo más habitual es disponer de varias sedes y tener que manejar información común, por lo tanto las bases de datos distribuidas son especialmente útiles.

**Fuentes:**

da Silva, B. (2015). *Arquitecturas de las BDD*. slideplayer.  
<https://slideplayer.es/slide/1075842/>

*Principles of Distributed Database Systems* (3ª ed.). (2011). Springer.

G. Rivas de la Fuente. "Cliente/servidor y Bases de Datos distribuidas. Implementación práctica de soluciones distribuidas". ComputerWorld | Innovación, negocio y tecnología.  
<https://www.computerworld.es/archive/clienteservidor-y-bases-de-datos-distribuidas-implimentacion-practica-de-soluciones-distribuidas> (accedido el 9 de marzo de 2022).

"Unidad 3: Sistemas de multibase de datos". Tópicos de base de datos.  
[https://karenkurapikadb.wordpress.com/2016/12/04/unidad-3sistemas-de-multibase-de-datos/#:~:text=Un%20sistema%20multibase%20de%20datos,base%20de%20datos%20\(SMBD\)](https://karenkurapikadb.wordpress.com/2016/12/04/unidad-3sistemas-de-multibase-de-datos/#:~:text=Un%20sistema%20multibase%20de%20datos,base%20de%20datos%20(SMBD)) (accedido el 9 de marzo de 2022).

"3.5 APLICACIONES MULTIBASE DE DATOS". TÓPICOS DE BASE DE DATOS.  
<https://cerbv08.wordpress.com/3-5-aplicaciones-multibase-de-datos/> (accedido el 9 de marzo de 2022).

NOTA: Las contribuciones de equipo están en rojo, las de Andrea están en color verde, las de Edgar en azul y las de Carlos en negro.