



Bases de Datos Distribuidas

Carlos Camacho Báñez

Juan Manuel Ruiz Pérez



INDICE

- 1. Diferencias entre BDD y BDC
- 2. ¿Cuándo es preferible usar una BDD?
- 3. Transparencia en BDD
- 4. Arquitecturas de referencia distribuidas
- 5. Estrategias de diseño de BDD
- 6. Técnicas de fragmentación
- 7. Procesamiento de una consulta en BDD
- 8. Funciones del administrador
- 9. Conclusiones
- 10. Bibliografía

DIFERENCIAS ENTRE BDD Y BDC

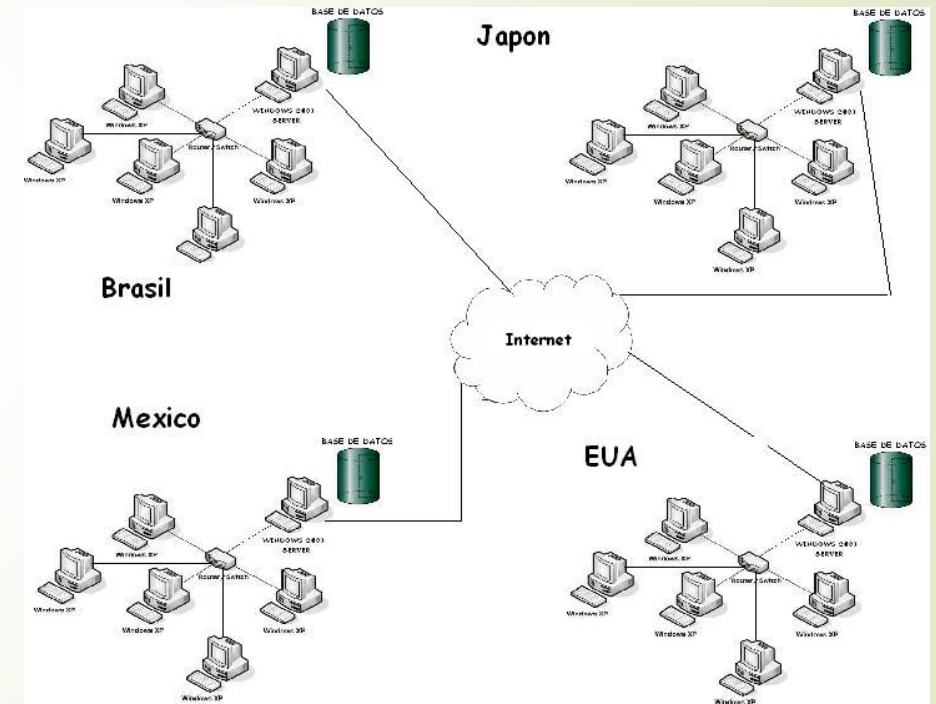
■ Base de Datos Centralizada (BDC):

- Un único servidor centralizado
- Acceso a través de la misma red desde diferentes usuarios
- Operaciones y transacciones se realizan en el mismo lugar
- En caso de fallo, se ve afectada el acceso a los datos para todos los usuarios



DIFERENCIAS ENTRE BDD Y BDC

- Base de Datos Distribuidas (BDD)
 - Almacenamiento en distintos servidores
 - Cada servidor tiene copias de datos parciales o completas
 - Comunicación entre servidores mediante una red para mantener la coherencia de datos.
 - No ha de ser la misma red para todos los usuarios





DIFERENCIAS ENTRE BDD Y BDC

Datos centralizados (BDC)

- ✓ Beneficiosos para la administración y control de datos
- ✓ Baja probabilidad de errores
- ✓ Baja sobrecarga de procesamiento
- x Servidor falla -> pérdida de datos e inactividad

Datos distribuidos (BDD)

- ✓ Mayor disponibilidad y redundancia
- ✓ Distribución de carga de trabajo -> mejora rendimiento
- ✓ Compartimiento de datos
- x Mayor dificultad de mantenimiento



¿CUANDO ES PREFERIBLE USARLA?

- Principalmente:

- Empresas o instituciones que no tienen su actividad centralizada, para tratar la base de datos de diferentes formas.
- Empresas o instituciones que requieren que la información se distribuya en varios nodos debido a su gran tamaño, para evitar ralentizaciones.
- Cuando se requiera de mayor escalabilidad, disponibilidad o tolerancia a fallos, o distribución o rendimiento.

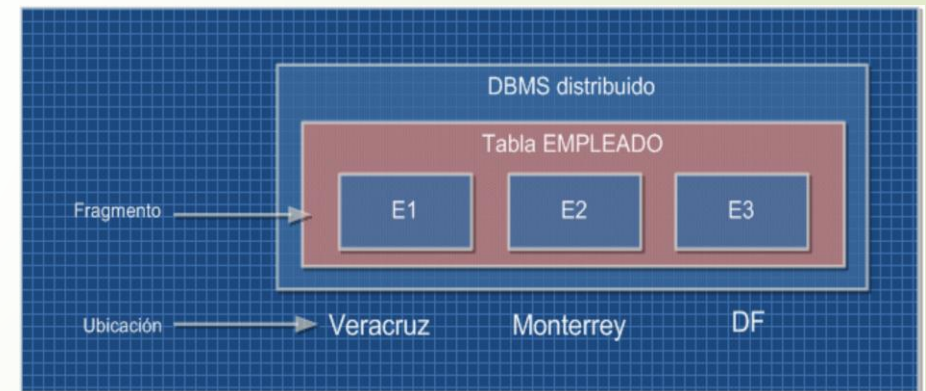


TRANSPARENCIA EN BDD

- Consiste en el grado hasta el cual los usuarios del sistema pueden ignorar los detalles del diseño distribuido.
- \uparrow Transparencia = \uparrow Ignorancia del usuario.
- Permite a los usuarios y las aplicaciones centrarse en el uso de los datos y no en la gestión de la base de datos distribuida.
- Mejora la eficiencia y la aparición de errores.
- Simplifica la gestión y el uso de datos.
- Algunos tipos de transparencia son: de distribución, de fragmentación, de replicación y de acceso.

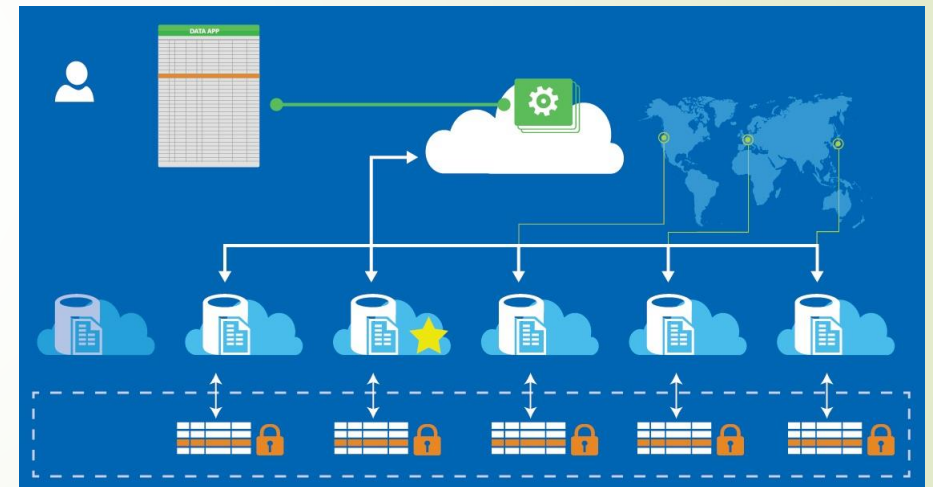
TRANSPARENCIA EN BDD

- Transparencia de distribución:
 - Visión usuarios: Acceso a los datos en un único servidor.
 - Realidad: Acceso a los datos en varios servidores.



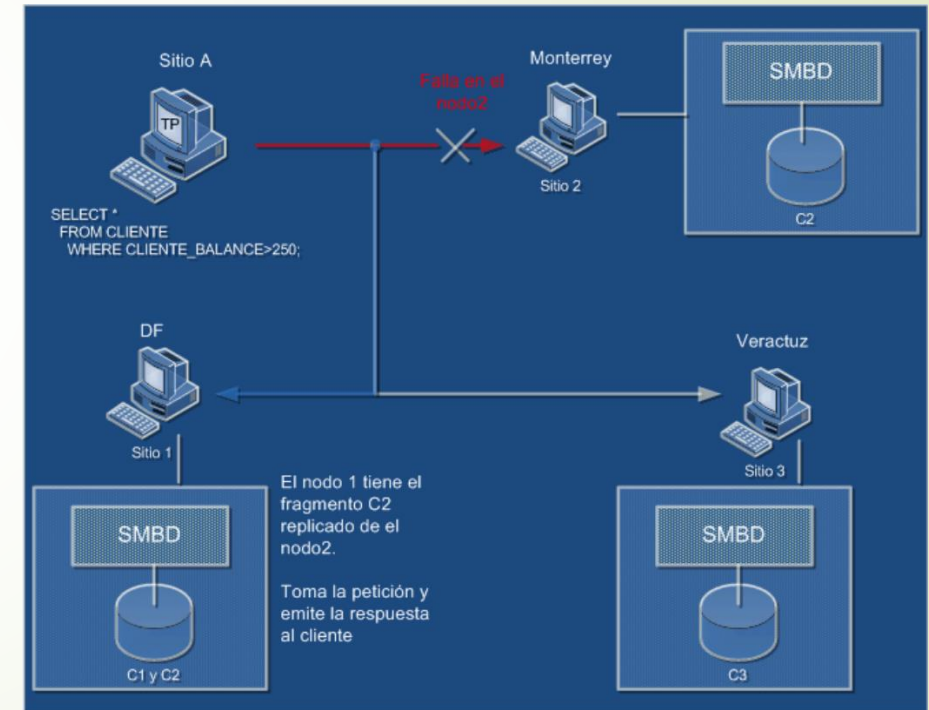
TRANSPARENCIA EN BDD

- Transparencia de fragmentación:
 - Visión usuarios: datos almacenados en una única tabla
 - Realidad: datos fragmentados en distintas tablas



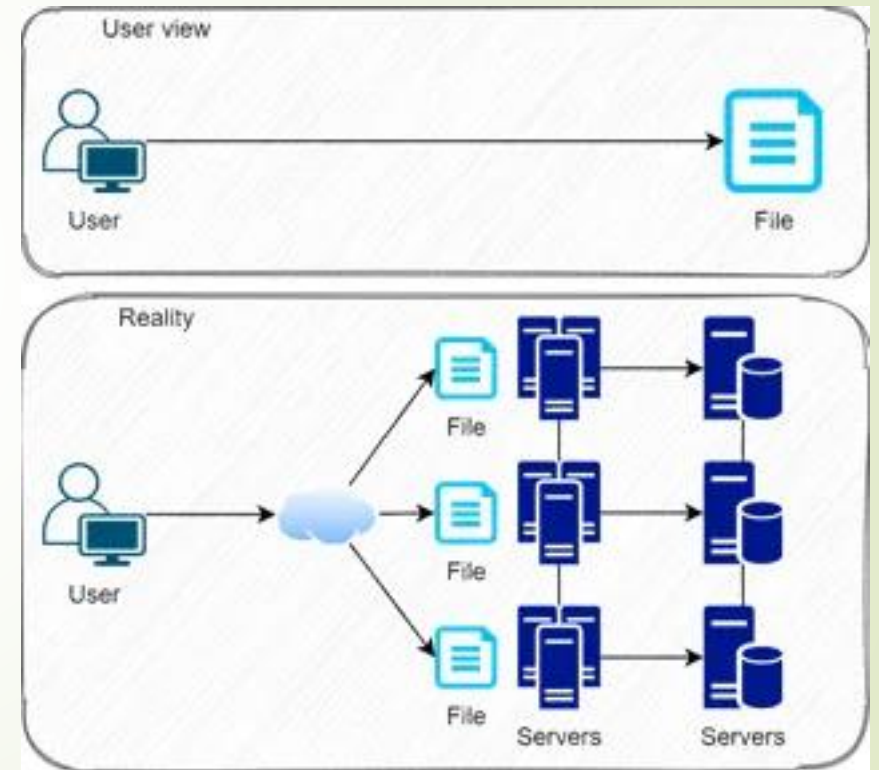
TRANSPARENCIA EN BDD

- Transparencia de replicación:
 - Visión usuarios: Datos almacenados en una única ubicación
 - Realidad: Datos replicados en varias ubicaciones para mejorar la redundancia y la disponibilidad



TRANSPARENCIA EN BDD

- Transparencia de acceso:
 - Visión usuario: No tiene preocupación de como acceder a los datos (elección de servidor, gestionar las conexiones, etc.)
 - Realidad: Administrador de la BDD es el encargado de realizar las tareas necesarias para garantizar dicha transparencia



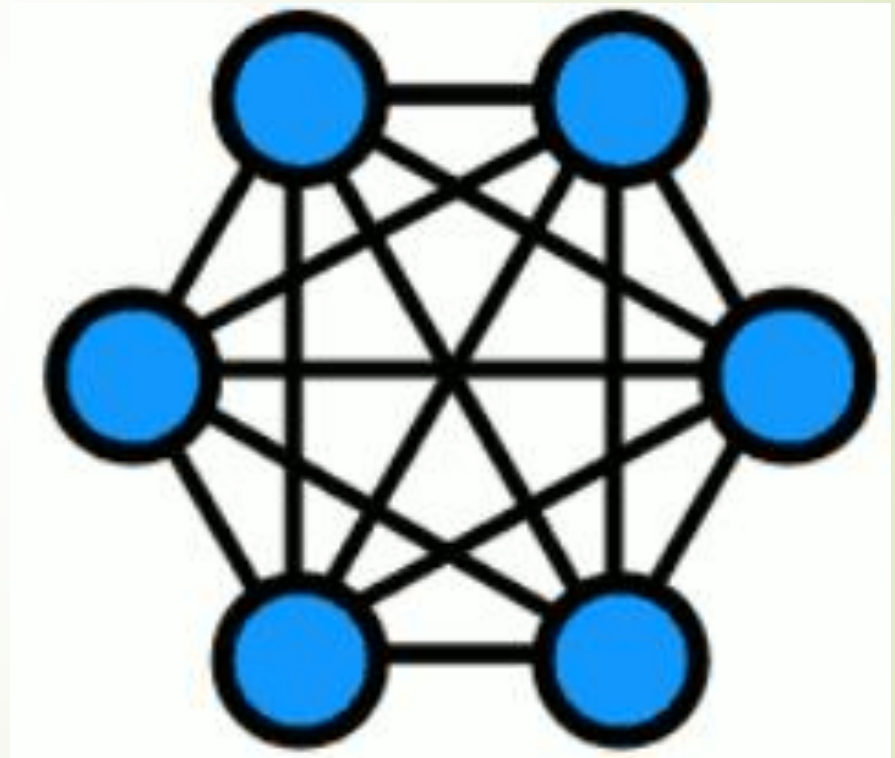


ARQUITECTURAS DE REFERENCIA

- Podemos organizarlas según:
 - Según su estructura física: el sistema se estructura según las conexiones de los equipos de sus subredes.
 - Según su grado de homogeneidad: el sistema se considera homogéneo o heterogéneo en función del parecido de los diferentes subsistemas.
 - Según el tipo de ambiente: el sistema se considera que sigue una arquitectura en función de lo que se comparta en él.

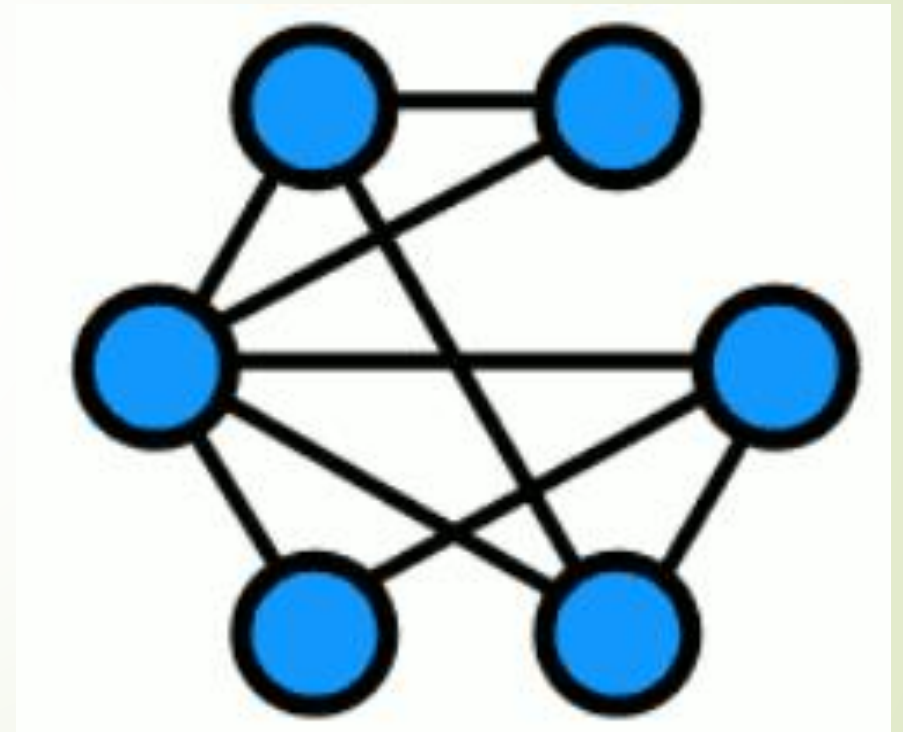
ARQUITECTURAS DE REFERENCIA SEGÚN SU ESTRUCTURA FÍSICA

- Red totalmente conectada
 - Los equipos se interconectan con el resto de equipos de la red.
 - La información se comparte de forma más efectiva.
 - Tiene mayor complejidad.



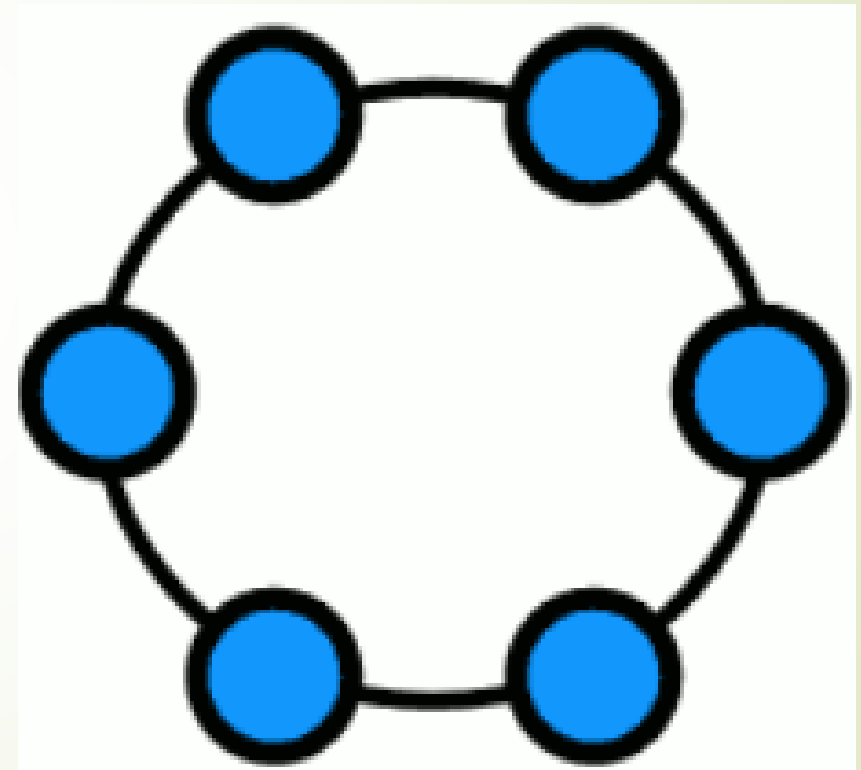
ARQUITECTURAS DE REFERENCIA SEGÚN SU ESTRUCTURA FÍSICA

- Red prácticamente conectada
 - Los equipos se interconectan con casi todos los equipos de la red.
 - Presenta menor tolerancia a fallos.
 - Tiene menor complejidad que la totalmente conectada.



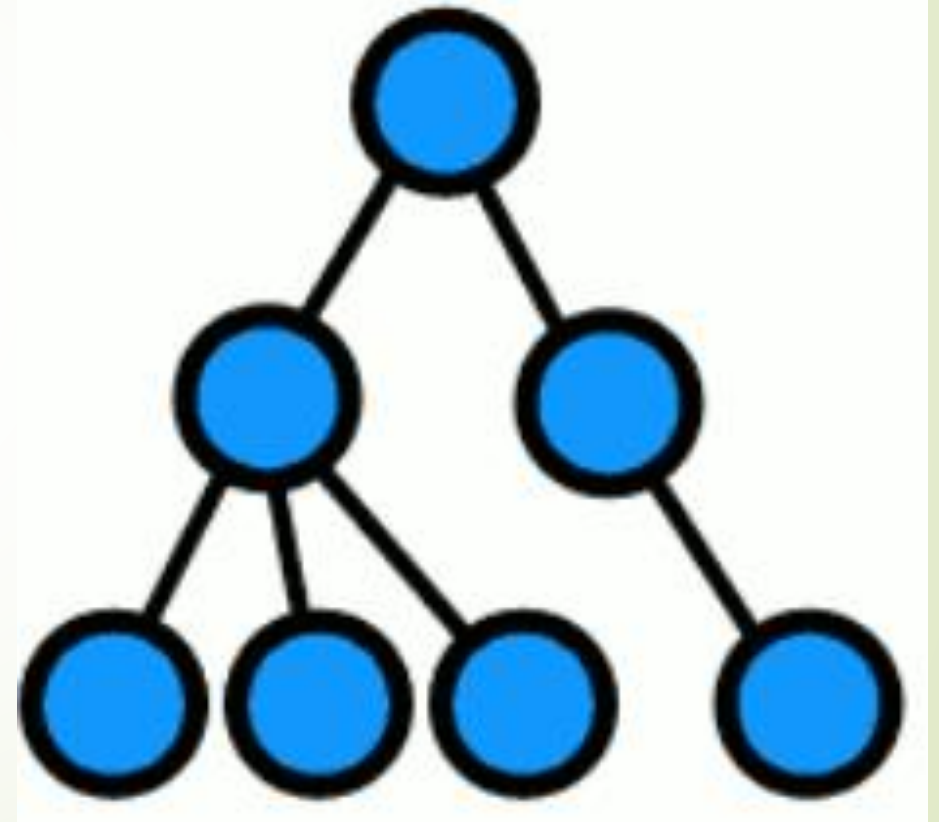
ARQUITECTURAS DE REFERENCIA SEGÚN SU ESTRUCTURA FÍSICA

- Red de anillo
 - Los equipos se interconectan con dos equipos "adyacentes".
 - Tiene una mayor eficiencia por la menor cantidad de información transmitida.
 - Genera más latencia por el volumen de información transmitida.



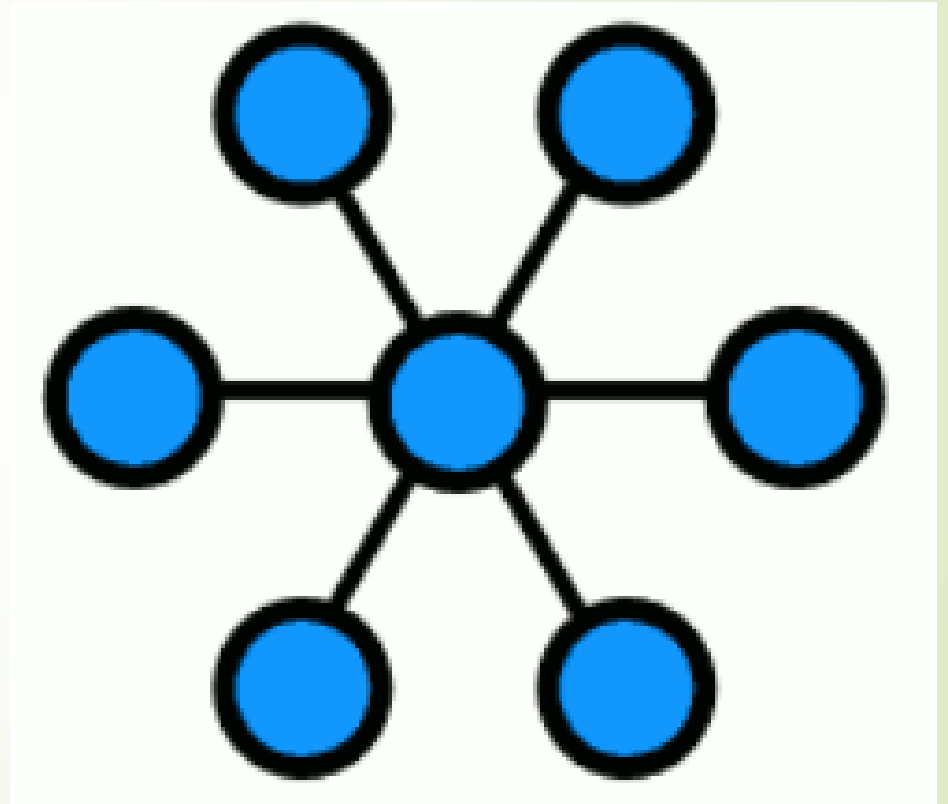
ARQUITECTURAS DE REFERENCIA SEGÚN SU ESTRUCTURA FÍSICA

- Red con estructura de árbol
 - Los equipos se conectan con un nodo padre, y este se conecta con el resto de padres.
 - Presenta gran escalabilidad por su misma estructura.
 - Tiene gran tolerancia a fallos, si un nodo falla el resto funciona.



ARQUITECTURAS DE REFERENCIA SEGÚN SU ESTRUCTURA FÍSICA

- Red de estrella
 - Todos los equipos se conectan a un único nodo central.
 - Tiene mayor eficiencia ya que se reduce la cantidad de flujo.
 - La latencia es baja, un equipo solo se comunica con el nodo central.





ARQUITECTURAS DE REFERENCIA SEGÚN SU HOMOGENEIDAD

- Sistemas homogéneos:
 - Los datos se distribuyen a través de los nodos.
 - Todos los nodos utilizan el mismo SGBD.
 - El SGBD administra todos los datos.
 - Todos los usuarios tienen acceso a la BD a través de la misma interfaz.

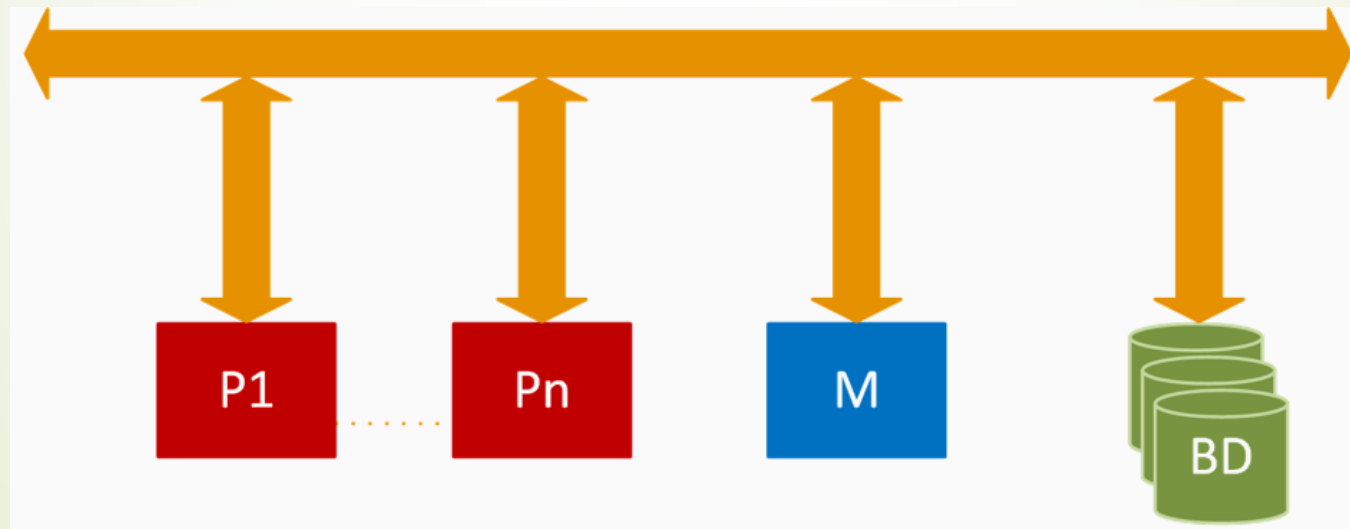


ARQUITECTURAS DE REFERENCIA SEGÚN SU HOMOGENEIDAD

- Sistemas heterogéneos:
 - Los datos se distribuyen a través de los nodos.
 - Los nodos utilizan SGBD diferentes según sus necesidades.
 - Algunos usuarios pueden requerir únicamente acceso local.
 - Existe un esquema local para que los usuarios puedan acceder a datos remotos.

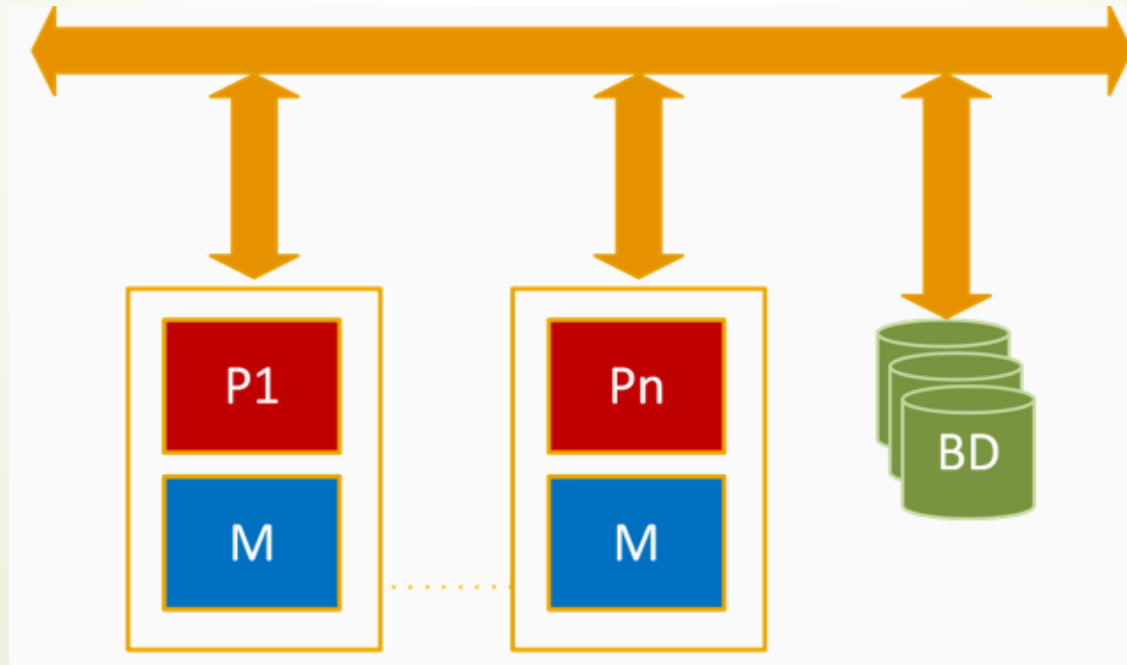
ARQUITECTURAS DE REFERENCIA SEGÚN EL TIPO DE AMBIENTE

- Memoria compartida: diversos procesadores que acceden a la misma memoria y a la misma unidad de almacenamiento.



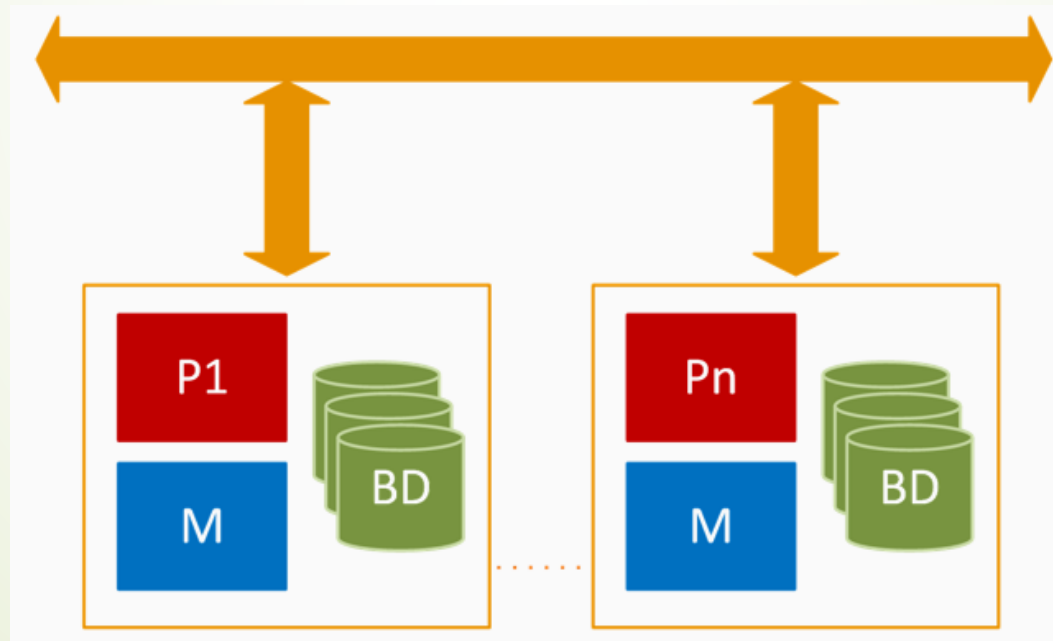
ARQUITECTURAS DE REFERENCIA SEGÚN EL TIPO DE AMBIENTE

- Disco compartida: diferentes procesadores que tienen su propia memoria local pero comparten unidad de almacenamiento.



ARQUITECTURAS DE REFERENCIA SEGÚN EL TIPO DE AMBIENTE

- Nada compartido: diferentes procesadores con diferente memoria y diferente unidad de almacenamiento.





ESTRATEGIAS DE DISEÑO DE BDD

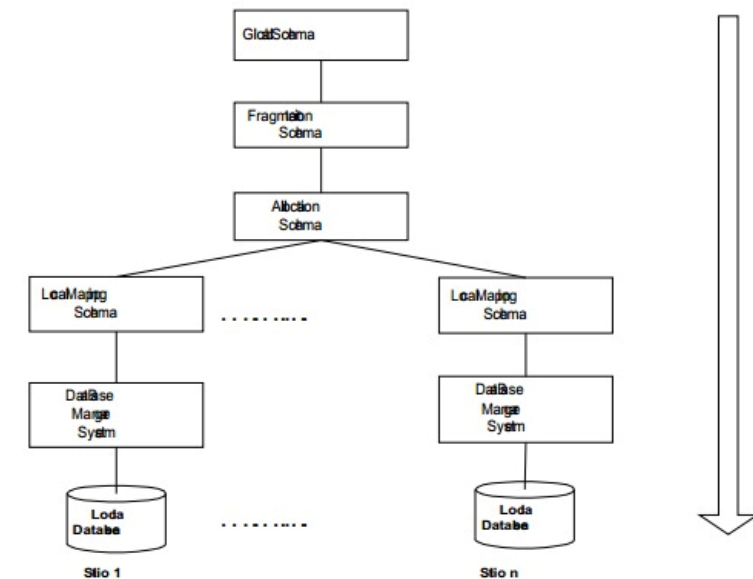
- Estructuras enfocadas al diseño de la BDD
- Son no excluyentes
- Se pueden utilizar ambas en distintas partes del diseño de la BDD
- Top-Down
- Bottom-Up

ESTRATEGIAS DE DISEÑO DE BDD

➤ Top-Down:

- Parte “desde 0”
- El diseñador conoce como se va a fragmentar y replicar los datos
- Control total del diseño
- Dificultad de mantenimiento a medida que crece la BD

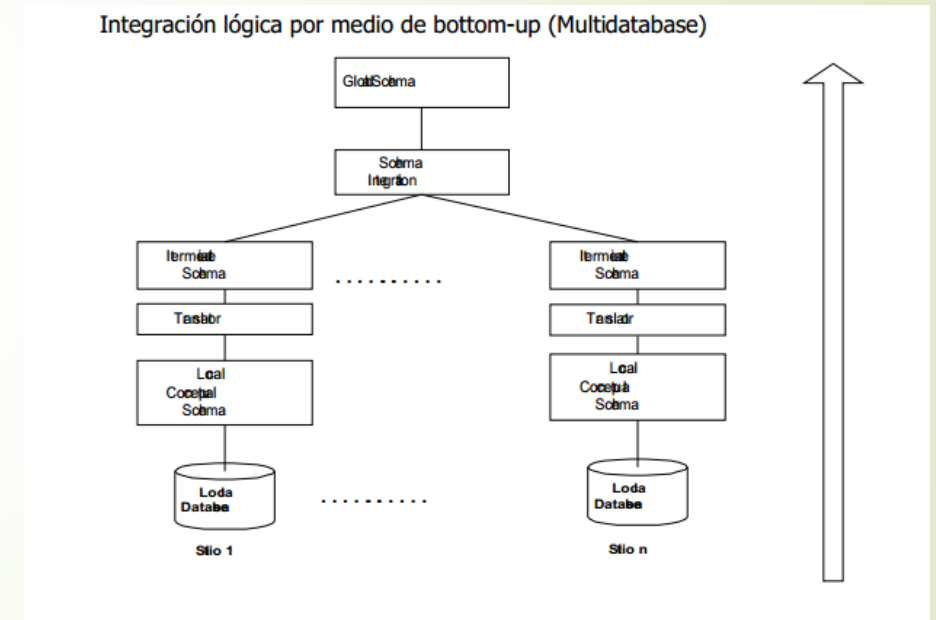
Integración lógica por medio de diseño top-down (DistDB)



ESTRATEGIAS DE DISEÑO DE BDD

➤ Bottom-Up

- Diseño a partir de BD existentes
- Define como se interconectan las BD locales para formar una BDD
- Enfoque más descentralizado
- Problemas de inconsistencia de datos y falta de coordinación entre las BD locales





TECNICAS DE FRAGMENTACION

- Existen 3 técnicas de fragmentación de una base de datos:
 - Fragmentación horizontal: Se divide por filas y cada conjunto de filas se almacena en un nodo.
 - Fragmentación vertical: Se divide por columnas y cada columna se almacena en un nodo diferente.
 - Fragmentación híbrida: Es una mezcla de las dos anteriores. La tabla se divide en filas y después en columnas.

TECNICAS DE FRAGMENTACION

FRAGMENTACION HORIZONTAL

S#	P#	DEPT#	CANT
S1	P1	D1	100
S1	P3	D1	100
S2	P1	D2	50
S2	P2	D2	200
S3	P3	D1	100
S4	P4	D2	50
S4	P5	D3	100
S4	P6	D1	50

S#	P#	DEPT#	CANT
S1	P1	D1	100
S1	P3	D1	100
S2	P1	D2	50
S2	P2	D2	200

S#	P#	DEPT#	CANT
S3	P3	D1	100
S4	P4	D2	50
S4	P5	D3	100
S4	P6	D1	50

TECNICAS DE FRAGMENTACION

FRAGMENTACION HORIZONTAL

EMP#	NOMBRE	SALARIO	IMPTO.	#JEFE	DEPT#
e1	X	1000	100	J1	D1
e2	Y	1500	300	J1	D1
e3	Z	500	20	J2	D2
e4	A	4000	1000	J3	D3
e5	B	2000	350	J2	D2

EMP#	NOMBRE	#JEFE	DEPT#
e1	X	J1	D1
e2	Y	J1	D1
e3	Z	J2	D2
e4	A	J3	D3
e5	B	J2	D2

EMP#	SALARIO	IMPTO
e1	1000	100
e2	1500	300
e3	500	20
e4	4000	1000
e5	2000	350

TECNICAS DE FRAGMENTACION

FRAGMENTACION HORIZONTAL

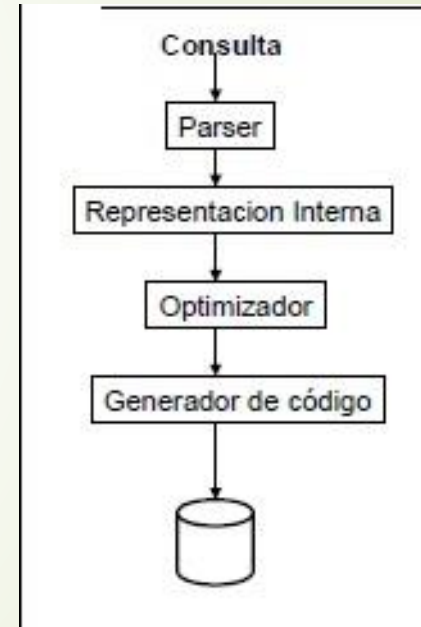
Escuela	Situación	Número alumnos
EUI	Campus sur	3000
EUIT	Campus sur	2800
TOPOGRAFIA	Campus sur	800
ETSIT	Ciudad Universitaria	2500
FI	Campus Montegancedo	2100

Escuela	Situación
EUI	Campus sur
EUIT	Campus sur
TOPOGRAFIA	Campus sur
ETSIT	Ciudad Universitaria
FI	Campus Montegancedo

Escuela	Número alumnos
EUI	3000
EUIT	2800
TOPOGRAFIA	800
ETSIT	2500
FI	2100

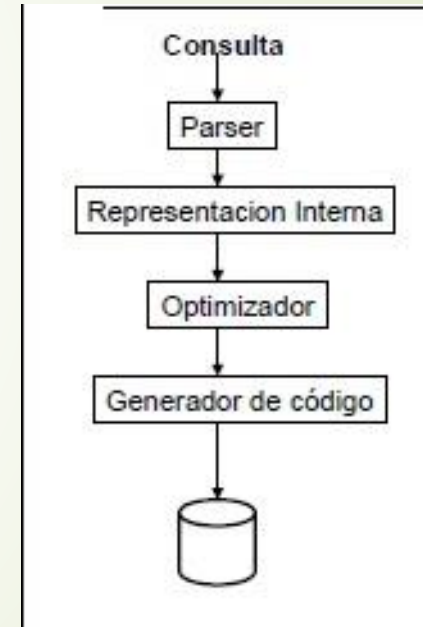
PROCESAMIENTO DE UNA CONSULTA EN BDD

- Análisis de la consulta:
 - Analizar la consulta y dividirla en diferentes partes (SELECT, INSERT, etc.)
 - Eliminación de redundancias, analizar semánticamente, etc.
- Optimización de la consulta:
 - Elegir el plan de ejecución mas adecuado para la consulta



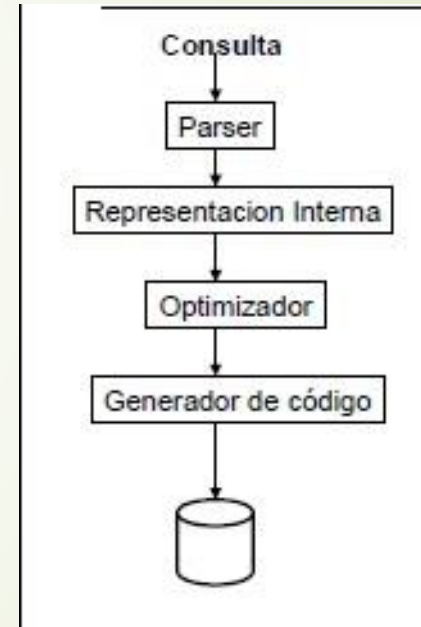
PROCESAMIENTO DE UNA CONSULTA EN BDD

- Distribución de la consulta:
 - Dividir en subconsultas
 - Distribuir a los diferentes nodos de la BDD
 - Cada subconsulta se ejecutará en los nodos relevantes
- Ejecución de la consulta:
 - Ejecución de cada subconsulta en su correspondiente nodo
 - Obtención de resultados parciales.



PROCESAMIENTO DE UNA CONSULTA EN BDD

- Combinación de los resultados:
 - Combinación de los resultados parciales en un resultado que se devuelve al usuario.





FUNCIONES DEL ADMINISTRADOR

- Creación y eliminación de bases de datos: define la estructura y conecta los nodos, además de eliminar los datos de forma segura.
- Configuración de las opciones de seguridad y auditoría: configura la autenticación y autorización de acceso, y configura el registro de eventos de auditoría.
- Integración de los esquemas: recoge los datos de todas las bases locales y los convierte en un esquema.
- Realización de procedimientos de backup: se asegura de la correcta realización de copias de seguridad.



FUNCIONES DEL ADMINISTRADOR

- Determinar la fragmentación de las relaciones: define qué datos se almacenan en cada nodo y cómo se replican estos.
- Aplicación de restricciones de integridad y seguridad: garantiza la privacidad de los datos almacenados.
- Definición de estándares: se asegura también de su cumplimiento y de que sean claros y fáciles de seguir.



CONCLUSION



- Las BDD son una herramienta poderosa pero que necesitan de una buena planificación.
- Ofrecen beneficios como mayor disponibilidad, fiabilidad, etc. Conllevan una serie de riesgos, como la complejidad de la gestión.
- Presenta varias arquitecturas y fragmentaciones de datos.
- Los administradores juegan un papel fundamental.



BIBLIOGRAFIA



- <http://www.slideshare.net/jguerra42/base-de-datos-distribuidas-presentation>
- <http://www.monografias.com/trabajos82/base-datos-distribuidas/base-datos-distribuidas.shtml>
- http://html.rincondelvago.com/bases-de-datos-distribuidas_1.html
- <https://www.uv.mx/personal/ermeneses/files/2017/03/BDAClase12-Repaso.pdf>
- <https://www.tecnologias-informacion.com/distribuidas.html>
- <http://base-datos-ruiz-salas-fl.blogspot.com/2015/02/arquitectura-de-base-de-datos.html>
- Documentación proporcionada por el profesor