

## **Unidad II. Conceptos de Sistema y de Arquitectura de BD**

M. en C. Euler Hernández Contreras

### **Contenido**

- 2.1 Modelos de datos (Conceptual, Lógico y Físico); Esquemas, instancias y estado de la BD.
- 2.2 Arquitectura de 3 niveles.
- 2.3 Independencia de Datos.
- 2.4 Arquitectura de los SGBD.
- 2.5 Arquitecturas cliente/servidor y centralizada para los SGBD.

### **Referencia Bibliográfica**

1. Michael V. Mannino. Administración de bases de datos, diseño y desarrollo de aplicaciones, Tercera Edición. Mc Graw Hill Interamericana, México 2007, 712 págs.
2. Date C. J. Introducción a los Sistemas de Bases de Datos, Séptima Edición. Pearson Educación de México, México 2001.
3. Hoffer A. Jeffrey, Prescott Mary B., Topi Heikki. Modern Database Management, Ninth Edition, Pearson/Prentice, Estados Unidos 2009.
4. Elmasri Ramez, Navathe Shamkant B. Fundamentos de Sistemas de Bases de Datos, Quinta Edición. Pearson/Addison Wesley, Madrid España 2007, págs. 988 ISBN: 978-84-7829-085-7
5. Ramakrishnan Raghu, Gehrke Johannes. Sistemas de Gestión de Bases de Datos, Tercera Edición. McGraw-Hill/Interamericana de España, Madrid España 2007, págs. 654 ISBN: 978-84-481-5638-1
6. Ricardo Catherine M. Bases de Datos. Mc Graw Hill, México D.F. 2009, págs. 642. ISBN: 978-970-10-7275-2

## 2. 1. Modelos de datos (Conceptual, Lógico y Físico); Esquemas, instancias y estado de la BD.

Una de las características fundamentales de los sistemas de bases de datos es que proporcionan cierto nivel de abstracción de datos, al ocultar las características sobre el almacenamiento físico que la mayoría de usuarios no necesita conocer. Los modelos de datos son el instrumento principal para ofrecer dicha abstracción.

Un *modelo de datos* es un conjunto de conceptos que sirven para describir la estructura de una base de datos: los datos, las relaciones entre los datos y las restricciones que deben cumplirse sobre los datos. Los modelos de datos contienen también un conjunto de operaciones básicas para la realización de consultas (lecturas) y actualizaciones de datos. Además, los modelos de datos más modernos incluyen conceptos para especificar comportamiento, permitiendo especificar un conjunto de operaciones definidas por el usuario.

Los modelos de datos se pueden clasificar dependiendo de los tipos de conceptos que ofrecen para describir la estructura de la base de datos.

- a) Los *modelos de datos de alto nivel, o modelos conceptuales*, disponen de conceptos muy cercanos al modo en que la mayoría de los usuarios percibe los datos.
- b) Los *modelos de datos de bajo nivel, o modelos físicos*, proporcionan conceptos que describen los detalles de cómo se almacenan los datos en el ordenador. Los conceptos de los modelos físicos están dirigidos al personal informático, no a los usuarios finales.
- c) Los *modelos lógicos*, se encuentran entre estos dos extremos, cuyos conceptos pueden ser entendidos por los usuarios finales, aunque no están demasiado alejados de la forma en que los datos se organizan físicamente. Los modelos lógicos ocultan algunos detalles de cómo se almacenan los datos, pero pueden implementarse de manera directa en un ordenador.

Los modelos conceptuales utilizan conceptos como *entidades, atributos y relaciones*. Una *entidad* representa un objeto o concepto del mundo real como, por ejemplo, un empleado de la empresa inmobiliaria o una oficina. Un *atributo* representa alguna propiedad de interés de una entidad como, por ejemplo, el nombre o el salario del empleado. Una *relación* describe una interacción entre dos o más entidades, por ejemplo, la relación de trabajo entre un empleado y su oficina.

Cada SGBD soporta un modelo lógico, siendo los más comunes el *relacional*, el *de red* y el *jerárquico*. Estos modelos representan los datos valiéndose de estructuras de registros, por lo que también se denominan *modelos orientados a registros*. Hay una nueva familia de modelos lógicos, son los *modelos orientados a objetos*, que están más próximos a los modelos conceptuales.

Los modelos físicos describen cómo se almacenan los datos en el ordenador: el formato de los registros, la estructura de los ficheros (desordenados, ordenados, etc.) y los métodos de acceso utilizados (índices, etc.).

A la descripción de una base de datos mediante un modelo de datos se le denomina *esquema de la base de datos*. Este esquema se especifica durante el diseño, y no es de esperar que se modifique a menudo. Sin embargo, los datos que se almacenan en la base de datos pueden cambiar con mucha frecuencia: se insertan datos, se actualizan, etc. Los datos que la base de datos contiene en un determinado momento se denominan *estado de la base de datos* u *ocurrencia de la base de datos*.

La distinción entre el esquema y el estado de la base de datos es muy importante. Cuando definimos una nueva base de datos, sólo especificamos su esquema al SGBD. En ese momento, el estado de la base de datos es el “*estado vacío*”, sin datos. Cuando se cargan datos por primera vez, la base de datos pasa al “*estado inicial*”. De ahí en adelante, siempre que se realice una operación de actualización de la base de datos, se tendrá un nuevo estado. El SGBD se encarga, en parte, de garantizar que todos los estados de la base de datos sean estados válidos que satisfagan la estructura y las restricciones especificadas en el esquema. Por lo tanto, es muy importante que el esquema que se especifique al SGBD sea correcto y se debe tener muchísimo cuidado al diseñarlo. El SGBD almacena el esquema en su catálogo o diccionario de datos, de modo que se pueda consultar siempre que sea necesario.

## 2.2 Arquitectura de 3 niveles (ANSI/SPARC).

Hay tres características importantes inherentes a los sistemas de bases de datos:

- a) La separación entre los programas de aplicación y los datos.
- b) El manejo de múltiples vistas por parte de los usuarios
- c) El uso de un catálogo para almacenar el esquema de la base de datos.

En 1975, el comité ANSI-SPARC (*American National Standard Institute Standards Planning and Requirements Committee*) propuso una arquitectura de tres niveles para los sistemas de bases de datos.

El objetivo de la arquitectura de tres niveles (o esquemas) es el de separar los programas de aplicación de la base de datos física. En esta arquitectura, el esquema de la base de datos se define en tres niveles de abstracción distintos (Ver Figura 2):

- a) En el *nivel interno* se describe la estructura física de la base de datos mediante un *esquema interno*. Este esquema se especifica mediante un modelo físico y describe todos los detalles para el almacenamiento de la base de datos, así como los métodos de acceso.
- b) El *nivel conceptual* se describe la estructura de toda la base de datos para una comunidad de usuarios (todos los de una empresa u organización), mediante un *esquema conceptual*. Este esquema oculta los detalles de las estructuras de almacenamiento y se concentra en describir entidades, atributos, relaciones, operaciones de los usuarios y restricciones. En este

nivel se puede utilizar un modelo conceptual o un modelo lógico para especificar el esquema.

- c) El *nivel externo o de vistas* incluye varios *esquemas externos o vistas de usuario*. Cada esquema externo describe la parte de la base de datos que interesa a un grupo de usuarios determinado, y oculta a ese grupo el resto de la base de datos. En este nivel podemos usar un modelo de datos de alto nivel o un modelo lógico para especificar los esquemas.

Hay que destacar que los tres esquemas no son más que descripciones de los mismos datos pero con distintos niveles de abstracción. Los únicos datos que existen realmente están a nivel físico, almacenados en un dispositivo como puede ser un disco. En un SGBD basado en la arquitectura de tres niveles, cada grupo de usuarios hace referencia exclusivamente a su propio esquema externo. Por lo tanto, el SGBD debe transformar cualquier petición expresada en términos de un esquema externo a una petición expresada en términos del esquema conceptual, y luego, a una petición en el esquema interno, que se procesará sobre la base de datos almacenada. Si la petición es una obtención (consulta) de datos, será preciso modificar el formato de la información extraída de la base de datos almacenada, para que coincida con la vista externa del usuario.

El proceso de transformar peticiones y resultados de un nivel a otro se denomina correspondencia o transformación. Estas correspondencias pueden requerir bastante tiempo, por lo que algunos SGBD no cuentan con vistas externas.

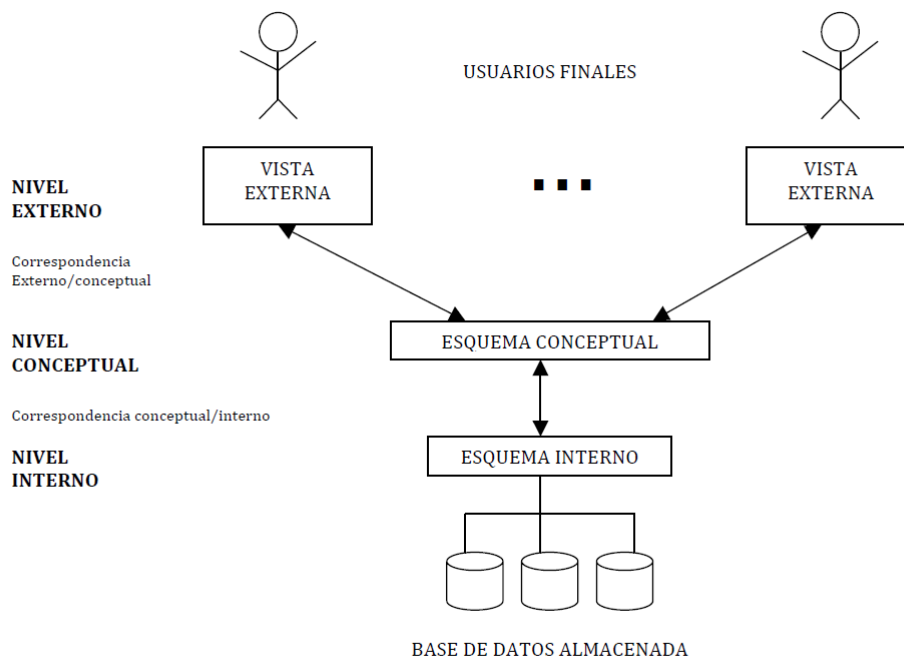


Figura 2. Arquitectura ANSI/SPARC

### 2.3 Independencia de Datos.

La arquitectura de tres esquemas puede servir para explicar el concepto de independencia de datos, que podemos definir como la capacidad para modificar

el esquema en un nivel del sistema de base de datos sin tener que modificar el esquema del nivel inmediato superior. Podemos definir dos tipos de independencia de datos:

1. La **independencia lógica de los datos** es la capacidad de modificar el esquema conceptual sin tener que alterar los esquemas externos ni los programas de aplicación. Podemos modificar el esquema conceptual para ampliar la base de datos (añadiendo un nuevo tipo de registro o un elemento de datos) o para reducir la base de datos (eliminando un tipo de registro o un elemento de datos). En el segundo caso, la modificación no deberá afectar a los esquemas externos que sólo se refieran a los datos restantes.
2. La **independencia física de los datos** es la capacidad de modificar el esquema interno sin tener que alterar el esquema conceptual (o los externos). Tal vez sea preciso modificar el esquema interno por la necesidad de reorganizar ciertos ficheros físicos (por ejemplo, al crear estructuras de acceso adicionales) con el fin de mejorar el rendimiento de las operaciones de recuperación y actualización. Si la base de datos aún contiene los mismos datos, no será necesario modificar el esquema conceptual.

## 2.4 Arquitectura de los SGBD.

La arquitectura de un SGBD típico basado en el modelo relacional de datos, presenta los siguientes elementos (Ver Figura 3):

### *Optimizador de consultas:*

Utiliza información sobre el modo en que se guardan los datos para producir un plan de ejecución eficiente para la evaluación de la consulta.

### *Gestor de memoria intermedia:*

Lleva las páginas del disco a la memoria principal según va haciendo falta, en respuesta a las solicitudes de lectura.

### *Gestor del espacio de disco:*

Se ocupa de la administración del espacio de disco, donde se almacenan los datos.

### *Gestor de transacciones:*

Garantiza que las transacciones soliciten y liberen los bloqueos de acuerdo con el correspondiente protocolo de bloqueo y programa la ejecución de las transacciones.

### *Gestor de bloqueos:*

Realiza un seguimiento de las solicitudes de bloqueo y concede los bloqueos sobre los objetos de la base de datos cuando quedan disponibles.

### Gestor de Recuperación:

Responsable del mantenimiento de un registro y de la restauración del sistema a un estado consistente tras los fallos.

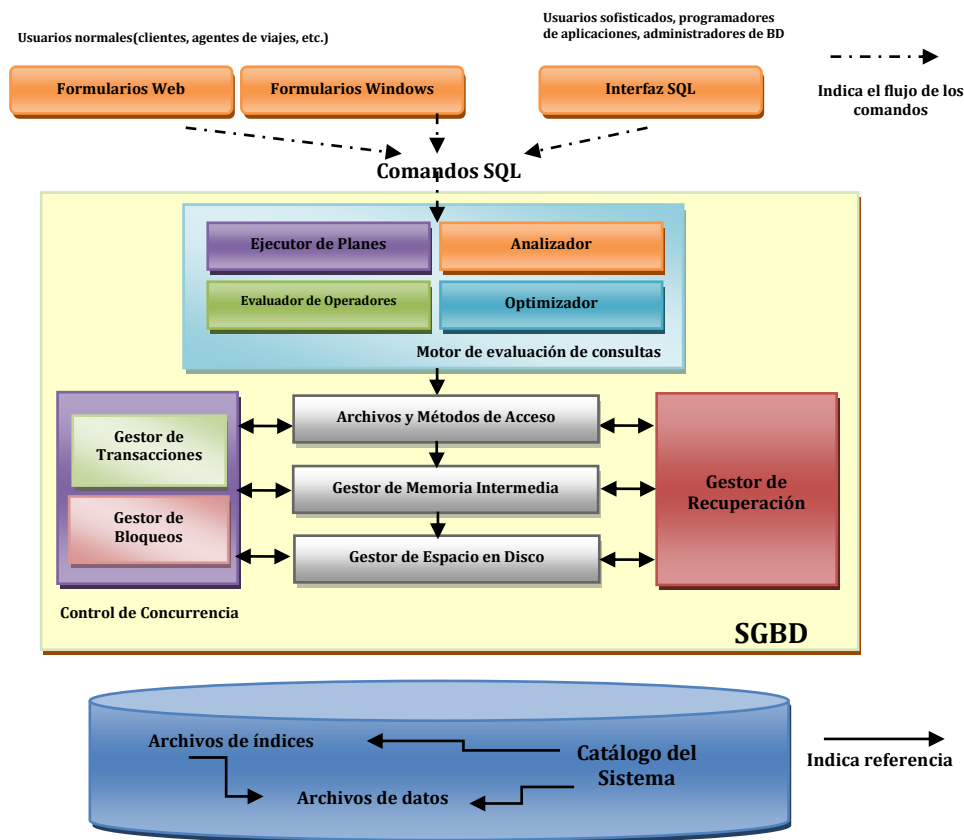


Figura 3. Arquitectura de un SGBD

## 2.5 Arquitecturas cliente/servidor y centralizada para los SGBD.

### Arquitectura Cliente/Servidor

Es un modelo general de interacción entre procesos software donde los procesos que interactúan están divididos en:

- Clientes (los cuales requieren servicios), y
- Servidores (aquellos que ofrecen los servicios).

Las funciones del cliente y del servidor están bien identificadas.

El uso de máquinas diferentes para el cliente y para el servidor es particularmente conveniente en el ambiente de las Bd:

- Máquina cliente: Orientada a la interacción con el usuario y al soporte a las herramientas de productividad (email, textos, hojas de cálculo, etc.).
- Máquinas servidor: Debe tener una gran memoria principal (para soportar la gestión de buffers) y una alta capacidad de disco (para almacenar la BD completa).

Las siguientes figuras muestran la arquitectura clásica cliente/servidor para bases de datos, donde la parte del CLIENTE, tenemos varias terminales que se conectan por medio de una red de computadoras o un medio de enlace (como la INTERNET) con la parte del SERVIDOR, el cual contiene el Servidor de Base de datos (SGBD), el cual proporciona el servicio de :

- A) Administración de los datos (Funciones del SGBD)
- B) Espacio de Almacenamiento (donde se hospeda la BD).

La arquitectura cliente/servidor puede presentarse en dos versiones:

- A) Arquitectura de 2 niveles (capas).

En esta arquitectura al igual que la centralizada; del lado del cliente hay varias terminales que acceden o se conectan con la parte del servidor, donde la parte del servidor existe un solo computador que tiene instalado el servidor de base de datos, el servidor de aplicaciones y el servidor web.

En la Figura 4, se muestra una arquitectura lógica cliente/servidor que ilustra la existencia clientes que solicitan uno o varios servicios.

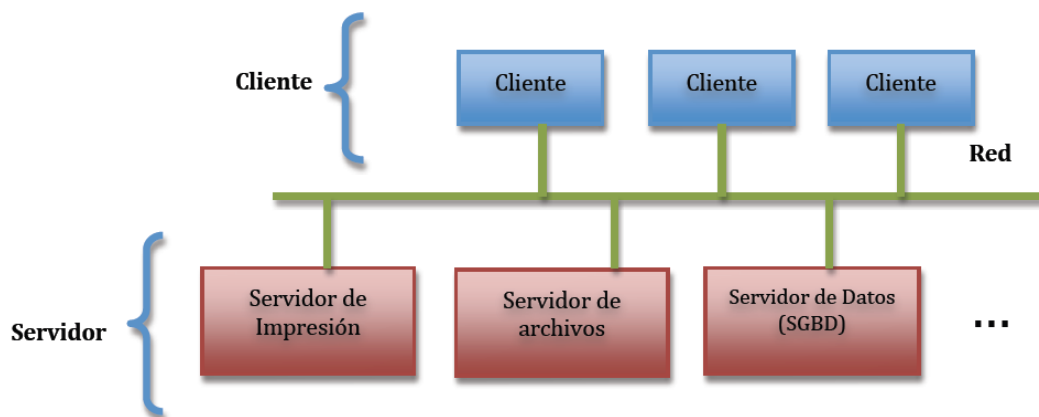


Figura 4. Arquitectura cliente/servidor de dos capas.

En la Figura 5, se muestra una arquitectura cliente/servidor física que ilustra como físicamente deben de implantarse dicha arquitectura.

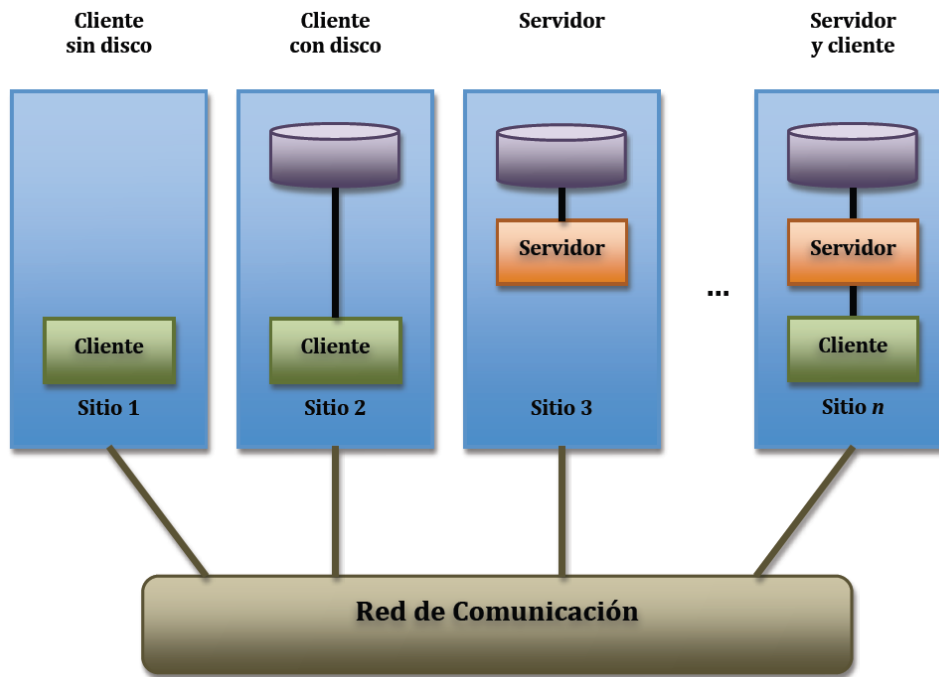


Figura 5. Arquitectura cliente/servidor física de dos capas.

#### B) Arquitectura de 3 niveles (capas).

Muchas aplicaciones web utilizan una arquitectura denominada de tres capas, que añade una capa intermedia entre el cliente y el servidor de la base de datos (Ver Figura 6).

Esta capa *intermedia* se denomina a veces *servidor de aplicaciones* y, en ocasiones, *servidor web*, en función de la aplicación. Este servidor juega un papel intermedio almacenando las reglas comerciales (procedimientos o restricciones) que se utilizan para acceder a los datos del servidor de base de datos. También puede mejorar la seguridad de la base de datos al comprobar las credenciales del cliente antes de enviar una solicitud al servidor de la BD.



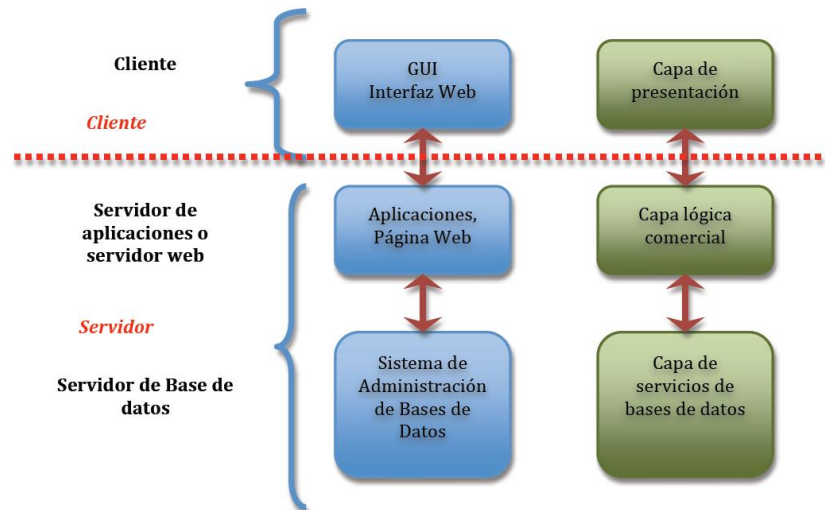


Figura 6. Arquitectura cliente/servidor lógica de tres capas.

c) Arquitectura de un Sistema de Bases de Datos Distribuidas

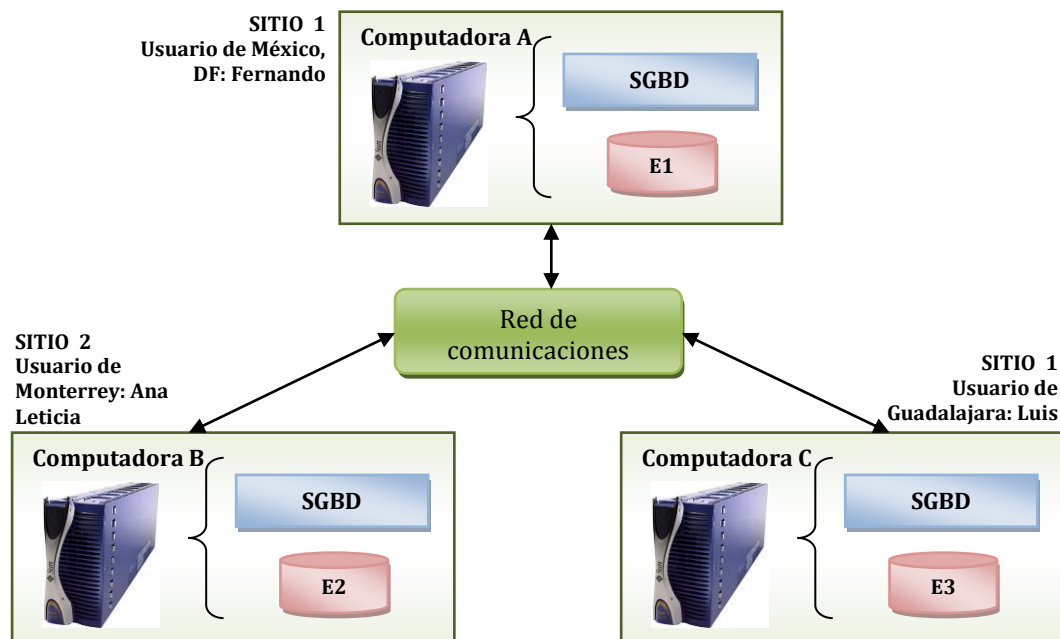


Figura 7. Arquitectura de una Base de Datos Distribuida.

La base de datos de la Figura 7 está dividida en tres fragmentos (E1, E2 y E3) localizados en diferentes sitios. Las computadoras están conectadas por medio de un sistema de red.

En una base de datos totalmente distribuida, los usuarios Fernando, Ana Leticia y Luis, no tienen que saber el nombre o ubicación de cada uno de los fragmentos para acceder la base de datos. Además los usuarios de otros sitios diferentes de México, DF., Guadalajara o Monterrey pueden acceder a la base de datos como una unidad lógica única.

La arquitectura de la Figura 7, se implementa usando el modelo de referencia de una base de datos distribuida (Ver Figura 8), el cual consta de las siguientes partes:

Esquema Global.

- Define los datos que contendrá un sistema de BDD como si la BD no fuera distribuida.
- El modelo de datos que es usado para definir el esquema global debería ser el conveniente a los sistemas de BD. (Modelo Relacional) -> define un conjunto de relaciones globales.

Esquema de Fragmentación

- Cada relación global puede ser dividido en diversas porciones llamados Fragmentos.
- Define un número de fragmentos a partir del esquema global.
- Fragmento: Son porciones lógicas de las relaciones globales, los cuales están físicamente localizados en uno o diferentes sitios de la red.

Esquema de localización.

- Define el sitio donde un fragmento es localizado.

