

*Equipo:*

- *Alpuche Morales Elmer Alexis*  
415075945 - *elmer.alpuche@ciencias.unam.mx*

- *Cova Pacheco Felipe de Jesús*  
312030111 - *felipejde.fc@ciencias.unam.mx*

## Tarea 1. Conceptos generales

### 1. Conceptos Generales

#### a) Indica las diferencias entre sistema de procesamiento de archivos y una base de datos relacional

Una manera de mantener información en un computador es hacerlo mediante un sistema de procesamiento de archivos típico o tradicional, que permitirá tener a los archivos estructurados y organizados, y poder realizar operaciones con ellos. Este sistema de archivos se mantiene mediante un sistema operativo convencional. Antes de la llegada de los sistemas de gestión de bases de datos (SGBD), las organizaciones normalmente han almacenado la información usando estos sistemas, pero mantener la información en estos sistemas de archivos tiene una serie de inconvenientes importantes:

- Redundancia e inconsistencia de datos. Existen datos que pueden repetirse en diferentes lugares o archivos, esto provoca que, teniendo esa duplicidad de datos, el almacenamiento y el costo (en recursos del sistema) de acceso sean más altos. Inconsistencia de datos se presentará porque las copias de los mismos datos en diferentes archivos pueden no coincidir, pues si en un archivo se hicieron cambios de los datos, en los otros archivos donde estaban los mismos datos no son modificados automáticamente.

- Dificultad en el acceso a los datos. Cuando se requiere de ciertos datos diferentes de archivos diferentes, la obtención, consulta y modificación de los datos no puede hacerse directamente de forma práctica y eficiente. Tendrían que desarrollarse sistemas de recuperación de datos para realizar esa operación específica, o desarrollar un sistema de recuperación de datos para uso general y ajustarlo de acuerdo a las necesidades.

- Aislamiento de datos. Debido a que los datos están dispersos en varios archivos, y los archivos pueden estar en diferentes formatos, es difícil escribir nuevos programas de aplicación para recuperar los datos apropiados.

- Problemas de integridad. Los valores de los datos almacenados en la BD deben satisfacer ciertas restricciones de consistencia. Los desarrolladores hacen cumplir estas restricciones en el sistema añadiendo código apropiado en las diversas aplicaciones. Sin embargo, cuando se añaden nuevas restricciones es difícil cambiar los programas para hacer que se cumplan. Esto se complica cuando las restricciones implican diferentes elementos de datos de diferentes archivos.

- Problemas de atomicidad. En muchas aplicaciones es crucial asegurar que, cuando ocurra un fallo y sea detectado, se restauren los datos a un estado de consistencia que existía antes del fallo. Es difícil asegurar esta propiedad en un sistema de archivos tradicional.

- Anomalías en el acceso concurrente. en estos sistemas un entorno en el que permita a múltiples usuarios actualizar los datos de un mismo archivo simultáneamente puede dar lugar a datos inconsistentes o un estado incorrecto.
- Problemas de seguridad. No todos los usuarios de un sistema de bases de datos deberían poder acceder a todos los datos. En estos sistemas es difícil garantizar tales restricciones de seguridad.

## **b) Ventajas y desventajas al trabajar con una base de datos**

### **i) Ventajas:**

- Almacenan grandes cantidades de información. Esto es muy útil para las grandes compañías que manejan grandes volúmenes de información
- Compartir la información. Los usuarios de distintas oficinas pueden compartir datos e información que son de gran importancia para sus departamentos o funciones
- Acceso rápido a la información. Esto es una gran ventaja, ya que la información siempre estará disponible para los usuarios
- Eliminación de información repetida o redundante. Los usuarios tendrán la certeza de que la información que están solicitando no está repetida o no es redundante
- Aumento en la productividad. Como la información está repetida y es coherente, los usuarios podrán aumentar su rendimiento al saber que cuentan con una información fiel.
- Reducción del espacio de estacionamiento. Al tener la información en medios electrónicos, se reduce considerablemente el espacio para almacenar de manera tradicional.
- Mejora la seguridad de la información. Existe un acceso reducido para los usuarios, de tal manera que cierta información podrá ser controlada por los administradores de la base de datos.
- Mejor mantenimiento. Al estar la información en medios electrónicos, el darle mantenimiento es mucho más fácil ya que se cuentan con herramientas para este mantenimiento

ii) Desventajas:

- Tamaño. Entre más grande sea la base de datos, se requiere mayor capacidad de disco duro y más memoria ram para que pueda funcionar adecuadamente
- Costo. El hardware y software para el correcto funcionamiento de una BDD es costoso
- Vulnerabilidad a los fallos. Esta es una gran desventaja, ya que las BDD están expuestas a fallos que no están en las manos de los usuarios, como una descarga eléctrica.

c) **Independencia de datos física y lógica, ¿cuál es más difícil de lograr?**

Lo podemos definir como la capacidad para modificar el esquema en un nivel del sistema sin tener que modificar el esquema del nivel inmediato superior.

i) Independencia de datos lógica:

Es la capacidad de modificar el esquema conceptual sin tener que alterar los esquemas externos ni los programas de aplicación.

Se puede modificar el esquema conceptual para ampliar la base de datos o para reducirla. Por ejemplo, al añadir cuentas de mercado de valores en un sistema bancario.

Son más difíciles de lograr que las independencias físicas, ya que los programas de aplicación son fuertemente dependientes de la estructura lógica de los datos a los que acceden

ii) Independencia de datos física:

Es la capacidad de modificar el esquema interno sin tener que alterar el esquema conceptual (o los externos).

Se refiere sólo a la separación entre las aplicaciones y las estructuras físicas de almacenamiento.

Por ejemplo, puede ser necesario reorganizar ciertos ficheros físicos con el fin de mejorar el rendimiento de las operaciones de consulta o de actualización de datos.

#### **d) Diccionario de datos e importancia**

Un diccionario de datos o repositorio de metadatos, es un repositorio centralizado de información sobre datos tales como significado, relación con otros datos, origen, uso y formato.

El diccionario de datos es un listado organizado de todos los datos que pertenecen a un sistema.

El objetivo de un diccionario de datos es dar precisión sobre los datos que se manejan en un sistema, evitando así malas interpretaciones o ambigüedades.

Si los analistas desean conocer cuántos caracteres abarca un determinado dato o qué otros nombres recibe en distintas partes del sistema, o dónde se utiliza, encontrarán las respuestas en un diccionario de datos desarrollado en forma apropiada.

#### **e) Indica las características de los siguientes modelos de bases de datos: jerárquico, de red, orientado a objetos**

##### **i) Jerárquico:**

Es un tipo de sistema de gestión de bases de datos que, como su nombre indica, almacena la información en una estructura jerárquica que enlaza los registros en forma de estructura de árbol (similar a un árbol visto al revés), en donde un nodo padre de información puede tener varios nodos hijo, y así sucesivamente.

Esta relación jerárquica no es estrictamente obligatoria, de manera que pueden establecerse relaciones entre nodos hermanos, y en este caso, la estructura en forma de árbol se convierte en una estructura en forma de gráfica dirigida.

##### **ii) De Red:**

Es una base de datos conformada por una colección o set de registros, los cuales están conectados entre si por medio de enlaces en una red. El registro es similar al de una red. El registro es similar al de una entidad como las empleadas en el modelo relacional.

Un registro es una colección o conjunto de campos (atributos), donde cada uno de ellos contiene solamente un único valor almacenado.

Una estructura de bases de datos de red, llamada algunas

veces estructura plex, abarca más que la estructura de árbol: un nodo hijo en la estructura red puede tener más de un nodo padre. En otras palabras, la restricción de que en un árbol jerárquico cada hijo puede tener sólo un padre, se hace menos severa. Así, la estructura de árbol se puede considerar como un caso especial de la estructura de red.

iii) Orientado a objetos:

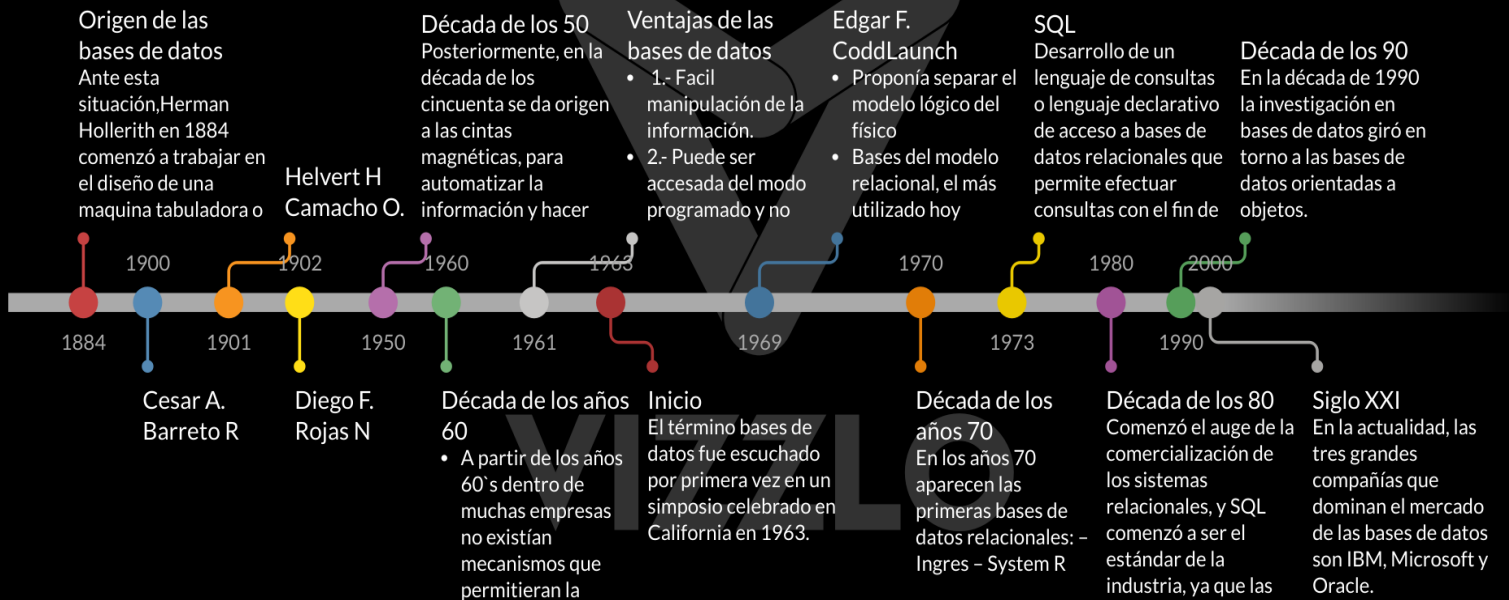
La información se representa mediante objetos como los presentes en la programación orientada a objetos. Cuando se integra las características de base de datos con las de un lenguaje de programación orientado a objetos, el resultado es un sistema gestor de base de datos orientada a objetos (ODBMS). Un ODBMS hace que las bases de datos aparezcan como datos de un lenguaje de programación en uno o más lenguajes de programación a los que dé soporte. Un ODBMS extiende los lenguajes con datos persistentes de forma transparente, control de concurrencia, recuperación de datos, consultas asociativas y otras capacidades.

Las bases de datos orientadas a objetos se diseñan bien para trabajar bien en conjunción con lenguajes de programación orientados a objetos, ya que los ODBMS usan exactamente el mismo modelo que estos lenguajes de programación.

## f) Línea del tiempo

# Desarrollo de las bases de datos

## Principales hitos



**g) Explica los conceptos de lenguaje para definición de datos, lenguaje para manipulación de datos, el modelo de datos, manejo de transacciones**

**a. Lenguaje para definición de datos:**

Un lenguaje para definición de datos (DDL por sus siglas en inglés) es un lenguaje proporcionado por el sistema de gestión de base de datos que permite a los programadores de la misma llevar a cabo tareas de definición de las estructuras que almacenan los datos así como de los procedimientos o funciones que permitan consultarlos.

Un lenguaje de descripción de datos es un lenguaje de programación para definir estructuras de datos

**b. Lenguaje para manipulación de datos:**

Es un lenguaje proporcionado por los sistemas gestores de bases de datos que permite a los usuarios de la misma llevar a cabo las tareas de consulta o modificación de los datos contenidos en las bases de datos del Sistema Gestor de Bases de Datos.

El lenguaje de manipulación de datos más popular hoy en día es SQL, usado para recuperar y manipular datos en una base de datos relacional, otros ejemplos de DML son los usados por bases de datos IMS/DL1, CODASYL u otras.

**c. Modelo de datos:**

Un modelo de datos es un lenguaje orientado a hablar de una Base de Datos. Típicamente un modelo datos permite describir:

- Las estructuras de datos de la base de datos. El tipo de datos que hay en la base y la forma en que se relacionan.
- Las restricciones de integridad. Un conjunto de condiciones que deben cumplir los datos para reflejar la identidad deseada.
- Operaciones de manipulación de los datos: típicamente, operaciones de agregado, borrado, modificación y recuperación de los datos de la base.



d. Manejo de transacciones:

- Una transacción en un sistema de gestión de bases de datos (SGBD) es un conjunto de ordenes que se ejecutan formando una unidad de trabajo, es decir, una forma invisible o atómica
- Transacción consiste en lograr hacer cualquier tipo de operación en una base de datos, basándonos en consultas desde las más simples hasta las de mayor grado de complejidad.
- Transacción se entiende en el ámbito de las bases de datos en lograr hacer acciones sobre las bases de datos deseadas, logrando operaciones de ingreso, borrado, actualización y visualizar.

## 2. Investigación

### a) Ciencia de datos y big data, ¿Cómo se relacionan con las BDD?

i) La ciencia de datos:

Es un campo disciplinario que involucra métodos científicos, procesos y sistemas para extraer conocimiento o un mejor entendimiento de datos en sus diferentes formas, ya sea estructurados o no estructurados, lo cual es una continuación de algunos campos de análisis de datos como la estadística, la minería de datos, el aprendizaje automático y la analítica predictiva. La ciencia de datos cumple con estas características:

- Trabaja de datos incompletos
- Los datos suelen estar desordenados
- Analiza los datos para ver qué información obtiene
- Grandes conjuntos de datos que es un desafío administrar
- Los hallazgos impulsan decisiones sobre operaciones y productos

iii) Big Data:

Cuando hablamos de Big Data nos referimos a conjuntos de datos o combinaciones de conjuntos de datos cuyo tamaño (volumen), complejidad (variabilidad) y velocidad

de crecimiento (velocidad) dificultan su captura, gestión, procesamiento o análisis mediante tecnologías y herramientas convencionales, tales como bases de datos relacionales y estadísticas convencionales o paquetes de visualización, dentro del tiempo necesario para que sean útiles

## **b) Bases de datos NoSQL:**

### **i) Bases de datos columnares:**

Estas son las bases de datos NoSQL más parecidas a las convencionales bases de datos relacionales. Almacenan datos estructurados en columnas individuales (en lugar de tablas).

Estas bases de datos utilizan grupos de columnas. Funcionan bien para datos generados por máquinas, fuentes de datos estructuradas demasiado grandes para ser manejadas por un solo ordenador, y para consultas rápidas de datos.

Si estás pensando en análisis rápido y de precisión de datos-máquina, estas pueden ser los tipos de base de datos ideales. Apache Cassandra y Apache HBase son algunas de ellas.

### **I ii) Bases de datos documentales:**

Estos tipos de bases de datos se basan en el almacenamiento de documentos en lugar de datos estructurados.

Son buenas para datos no estructurados, como texto abierto de una carta o de un correo electrónico, y para datos semi-estructurados como documentos académicos.

Tendrás que fijarte en ellas si estas pensando en análisis de texto de documentos demasiado grandes para las bases de datos convencionales. Algunas de las más conocidas son MongoDB y Apache Couch DB.

### **iii) Bases de datos gráficas:**

Estos tipos de bases de datos utilizan una estructura gráfica que es esencialmente un diagrama de las relaciones dentro de los datos, en lugar de tablas.

Son buenos motores de bases de datos para impulsar aplicaciones web que deban proporcionar información muy rápidamente, como las que se utilizan para las compras online y las plataformas de redes sociales.

Necesitarás mirar estos tipos de bases de datos si tu interés principal es una aplicación rápida, y puedes vivir con algunas aproximaciones en análisis.

Algunas de las más conocidas son Neo4J de Neo Technology's y Microsoft Horton.

iv) **Key-Value:**

Estas están diseñadas para desarrollo simple y fácil de aplicaciones.

Son buenas para situaciones donde necesitas trabajar con aplicaciones que se puedan desarrollar rápidamente y donde todas las demás consideraciones son secundarias.

Algunas de las más conocidas son Basho Technologies' Riak y Redis.

v) **XML:**

Estos tipos de bases de datos utilizan el lenguaje XML, el cual es el lenguaje subyacente de la Web y de otros muchos sistemas de intercambio de información, para definir la estructura de datos.

Son buenas para la gestión de datos que no se puede obtener con cualquier otro tipo de bases de datos, y un buen partido cuando se tiene una gran cantidad de datos en formatos no tradicionales, como vídeo y audio.

Tendrás que mirar este tipos de bases de datos cuando necesites profundizar en el análisis de datos no estructurados como analíticas de voz o de vídeo.

Algunos grandes nombres en estos tipos de bases de datos son Mark Logic y Sedna.

### Fuentes:

- [https://www.ibm.com/developerworks/ssa/data/library/tipos\\_bases\\_de\\_datos/index.html](https://www.ibm.com/developerworks/ssa/data/library/tipos_bases_de_datos/index.html)
- <http://es.ccm.net/contents/66-introduccion-a-las-bases-de-datos>
- <http://biblioteca.cinvestav.mx/index.php/bases-de-datos>
- <http://www.tiposde.org/informatica/139-tipos-de-base-de-datos/>
- <http://www.tecnologias-informacion.com/cgi-sys/suspendedpage.cgi>