



# INSTITUTO DE ESTUDIOS SUPERIORES DEL PACIFICO SUR A.C



**MATERIA: Administración De Base De Datos.**

**Licenciatura: Ingeniería En Sistemas Computacionales.**

**Nombre del alumno: Garcia Ramirez, Oscar Yosimar.**

**Nombre del profesor: Ing. Almicar Serrano Avila.**

**GRUPO: ÚNICO**

**FECHA: 25/07/2020**

# Índice

COMO FUNCIONAN LA BASE DE DATOS NO RELACIONALES.....	<b>3</b>
Y POR QUÉ REPRESENTAN UNA ALTERNATIVA CON RESPECTO A LAS RELACIONES .....	<b>6</b>



## **¿CÓMO FUNCIONAN LA BASE DE DATOS NO RELACIONALES?.**

### **¿Qué es una base de datos relacional?**

Una base de datos relacional es un tipo de base de datos que almacena y proporciona acceso a puntos de datos relacionados entre sí. Las bases de datos relacionales se basan en el modelo relacional, una forma intuitiva y directa de representar datos en tablas. En una base de datos relacional, cada fila de la tabla es un registro con un ID único llamado clave. Las columnas de la tabla contienen atributos de los datos, y cada registro generalmente tiene un valor para cada atributo, lo que facilita el establecimiento de las relaciones entre los puntos de datos.

### **¿Cómo se estructuran las bases de datos relacionales?**

El modelo relacional significa que las estructuras lógicas de datos—las tablas de datos, vistas e índices—están separadas de las estructuras físicas de almacenamiento. Esta separación significa que los administradores de bases de datos pueden administrar el almacenamiento físico de datos sin afectar el acceso a esos datos como una estructura lógica. Por ejemplo, cambiar el nombre de un archivo de base de datos no cambia el nombre de las tablas almacenadas en él.

La distinción entre lógica y física también se aplica a las operaciones de la base de datos, que son acciones claramente definidas que permiten a las aplicaciones manipular los datos y las estructuras de la base de datos. Las operaciones lógicas permiten que una aplicación especifique el contenido que necesita, mientras que las operaciones físicas determinan cómo se debe acceder a esos datos y luego realizan la tarea.

Para garantizar que los datos sean siempre precisos y accesibles, las bases de datos relacionales siguen ciertas reglas de integridad. Por ejemplo, una regla de integridad puede especificar que no se permitan filas duplicadas en una tabla, para eliminar la posibilidad de que ingrese información errónea en la base de datos.

### **El modelo relacional:**

En los primeros años de las bases de datos, cada aplicación almacenaba datos en su propia estructura única. Cuando los desarrolladores querían crear aplicaciones para usar esos datos, tenían que saber mucho sobre la estructura de datos particular para encontrar los datos que necesitaban. Estas estructuras de datos eran ineficientes, difíciles de mantener y difíciles de optimizar para ofrecer un buen rendimiento de la aplicación. El modelo de base de datos relacional se diseñó para resolver el problema de varias estructuras de datos arbitrarias. El modelo relacional proporcionó una forma estándar de representar y consultar datos que cualquier aplicación podría utilizar. Desde el principio, los desarrolladores reconocieron que la principal fortaleza del modelo de base de datos relacional estaba en el uso de tablas, que eran una forma intuitiva, eficiente y flexible de almacenar y acceder a información estructurada.

Con el tiempo, cuando los desarrolladores comenzaron a utilizar el lenguaje de consulta estructurado (SQL) para escribir y consultar datos en una base de datos, surgió otra fortaleza del modelo relacional. Durante muchos años, se utilizó ampliamente el SQL como lenguaje para consultas de bases de datos. El SQL, que se basa en el álgebra relacional, proporciona un lenguaje matemático internamente consistente que facilita la mejora del rendimiento de todas las consultas de la base de datos. En comparación, otros enfoques deben definir consultas individuales.

### **Beneficios de las bases de datos relacionales:**

Las organizaciones de todo tipo y tamaño utilizan el modelo relacional simple pero poderoso para una amplia variedad de necesidades de información. Las bases de datos relacionales se utilizan para hacer seguimiento de los inventarios, procesar transacciones de comercio electrónico, administrar grandes cantidades de información de clientes de misión crítica y mucho más. Se puede considerar una base de datos relacional para cualquier necesidad de información en la que los puntos de datos se relacionen entre sí y se deban administrar de una manera segura, consistente y basada en reglas.

Las bases de datos relacionales existen desde la década de 1970. Actualmente, las ventajas del modelo relacional continúan convirtiéndolo en el modelo más aceptado para bases de datos.

### **Consistencia de los datos:**

El modelo relacional es el mejor para mantener la consistencia de los datos en todas las aplicaciones y copias de la base de datos (denominadas *instancias*). Por ejemplo, cuando un cliente deposita dinero en un cajero automático y, luego, mira el saldo de la cuenta en un teléfono móvil, el cliente espera ver que ese depósito se refleje inmediatamente en un saldo de cuenta actualizado. Las bases de datos relacionales se destacan en este tipo de consistencia de datos, lo que garantiza que múltiples instancias de una base de datos tengan los mismos datos todo el tiempo.

Es difícil para otros tipos de bases de datos mantener este nivel de coherencia oportuna con grandes cantidades de datos. Algunas bases de datos recientes, como NoSQL, solo pueden proveer “consistencia eventual.” Bajo este principio, cuando la base de datos se escala o cuando varios usuarios acceden a los mismos datos al mismo tiempo, los datos necesitan algo de tiempo para “ponerse al día.” La consistencia eventual es aceptable para algunos usos, como para mantener listados en un catálogo de productos, pero para operaciones comerciales críticas como transacciones de un carrito de compras, la base de datos relacional sigue siendo el estándar de oro.

### **Compromiso y atomicidad:**

Las bases de datos relacionales manejan las reglas y políticas comerciales en un nivel muy detallado, con políticas estrictas sobre el *compromiso* (es decir, hacer un cambio permanente en la base de datos). Por ejemplo, considere una base de datos de inventario que hace seguimiento

de tres partes que siempre se usan juntas. Cuando se extrae una parte del inventario, las otras dos también se deben extraer. Si una de las tres partes no está disponible, ninguna de las partes debe quitarse—las tres partes deben estar disponibles antes de que la base de datos se comprometa. Una base de datos relacional no se comprometerá por una parte hasta que sepa que puede comprometerse por las tres. Esta capacidad de compromiso multifacética se llama *atomicidad*. La atomicidad es la clave para mantener la precisión de los datos en la base de datos y garantizar que cumpla con las reglas, regulaciones y políticas de la empresa.

### **Procedimientos almacenados y bases de datos relacionales:**

El acceso a los datos implica muchas acciones repetitivas. Por ejemplo, una consulta simple para obtener información de una tabla de datos puede necesitar repetirse cientos o miles de veces para producir el resultado deseado. Estas funciones de acceso a los datos requieren algún tipo de código para acceder a la base de datos. Los desarrolladores de aplicaciones no desean escribir un código nuevo para estas funciones en cada aplicación nueva. Afortunadamente, las bases de datos relacionales permiten procedimientos almacenados, que son bloques de código a los que se puede acceder con una simple llamada de aplicación. Por ejemplo, un solo procedimiento almacenado puede proporcionar un etiquetado de registro consistente para usuarios de varias aplicaciones. Los procedimientos almacenados también pueden ayudar a los desarrolladores a garantizar que ciertas funciones de datos en la aplicación se implementen de una manera específica.

### **Bloqueo de bases de datos y concurrencia:**

Pueden surgir conflictos en una base de datos cuando varios usuarios o aplicaciones intentan cambiar los mismos datos al mismo tiempo. Las técnicas de bloqueo y concurrencia reducen la posibilidad de conflictos mientras mantienen la integridad de los datos.

El *bloqueo* evita que otros usuarios y aplicaciones accedan a los datos mientras se actualizan. En algunas bases de datos, el bloqueo se aplica a toda la tabla, lo que crea un impacto negativo en el rendimiento de la aplicación. Otras bases de datos, como las bases de datos relacionales de Oracle, aplican bloqueos a nivel de registro, lo que deja disponibles los otros registros dentro de la tabla, lo que ayuda a garantizar un mejor rendimiento de la aplicación.

La *concurrencia* gestiona la actividad cuando varios usuarios o aplicaciones realizan consultas al mismo tiempo en la misma base de datos. Esta capacidad proporciona el acceso correcto a los usuarios y las aplicaciones de acuerdo con las políticas definidas para el control de datos.

### **¿Qué buscar a la hora de seleccionar una base de datos relacional?**

El software que se utiliza para almacenar, administrar, consultar y recuperar datos almacenados en una base de datos relacional se denomina *sistema de gestión de bases de datos relacionales (RDBMS)*. El RDBMS proporciona una interfaz entre usuarios y aplicaciones y la base de datos,

así como funciones administrativas para administrar el almacenamiento, el acceso y el rendimiento de los datos.

Varios factores pueden guiar su decisión al momento de elegir entre tipos de bases de datos y productos de bases de datos relacionales. El RDBMS que elija dependerá de las necesidades de su negocio. Hágase las siguientes preguntas:

- **¿Cuáles son nuestros requisitos de precisión de datos?** ¿El almacenamiento de datos y la precisión dependerán de la lógica empresarial? ¿Nuestros datos tienen requisitos estrictos de precisión (por ejemplo, datos financieros e informes gubernamentales)?
- **¿Necesitamos escalabilidad?** ¿Cuál es la escala de los datos a administrar y cuál es su crecimiento previsto? ¿Será necesario que el modelo de base de datos admita copias de base de datos duplicadas (como instancias separadas) para la escalabilidad? Si es así, ¿puede mantener la consistencia de los datos en esas instancias?
- **¿Qué tan importante es la concurrencia?** ¿Varios usuarios y aplicaciones necesitarán un acceso simultáneo a los datos? ¿El software de la base de datos admite concurrencia mientras protege los datos?
- **¿Cuáles son nuestras necesidades de rendimiento y confiabilidad?** ¿Necesitamos un producto de alto rendimiento y alta confiabilidad? ¿Cuáles son los requisitos para el rendimiento de la consulta-respuesta? ¿Cuáles son los compromisos de los proveedores para los acuerdos de nivel de servicio (SLA) o tiempo de inactividad no planificado?

## Y POR QUÉ REPRESENTAN UNA ALTERNATIVA CON RESPECTO A LAS RELACIONES:

Se ha dicho que las bases de datos no relacionales están "broken by ... no relacionales ha representado una alternativa distinta respecto a sus ... y entidad-relación como herramientas para modelado de bases de datos.

student_id	age	score
1	12	77
2	12	68
3	11	75



```
[
  {
    "student_id":1,
    "age":12,
    "score":77
  },
  {
    "student_id":2,
    "age":12,
    "score":68
  },
  {
    "student_id":3,
    "age":11,
    "score":75
  }
]
```