

PROGRAMA DE ESTUDIOS

DESARROLLO DE SISTEMAS DE INFORMACIÓN

**DISEÑO E IMPLEMENTACIÓN
DE BASE DE DATOS**

Tema

MODELOS DE DATOS.

1. Modelos de datos.

Un modelo de datos (colección de conceptos que se pueden utilizar para describir la estructura de una base de datos) proporciona los medios necesarios para conseguir esa abstracción. Por estructura de una base de datos nos referimos a los tipos de datos, relaciones y restricciones que deben mantenerse para los datos. La mayoría de modelos de datos también incluyen un conjunto de operaciones básicas para especificar las recuperaciones y actualizaciones en la base de datos.

2. Niveles de abstracción de una Base de Datos

Se han propuesto muchos modelos de datos, que podemos clasificar conforme a los tipos de conceptos que utilizan para describir la estructura de la base de datos. Los modelos de datos de alto nivel o conceptuales ofrecen conceptos muy cercanos a como muchos usuarios perciben los datos, mientras que los modelos de datos de bajo nivel o físicos ofrecen conceptos que describen los detalles de cómo se almacenan los datos en el computador. Los conceptos ofrecidos por los modelos de datos de bajo nivel están pensados principalmente para los especialistas en computadores, no para los usuarios finales normales. Entre estos dos extremos hay una clase de modelos de datos representativos (o de implementación),⁴ que ofrecen conceptos que los usuarios finales pueden entender pero que no están demasiado alejados de cómo se organizan los datos dentro del computador. Los modelos de datos representativos ocultan algunos detalles relativos al almacenamiento de los datos, pero pueden implementarse directamente en un computador. Los modelos de datos conceptuales utilizan conceptos como entidades, atributos y relaciones. Una entidad representa un objeto o concepto del mundo real, como un empleado o un proyecto que se describe en la base de datos. Un atributo representa alguna propiedad de interés que describe a una entidad, como, por ejemplo, el nombre o el salario de un empleado. Una relación entre dos o más entidades representa una asociación entre dos o más entidades; por ejemplo, una relación de trabajo entre un empleado y un proyecto.

3. Modelo conceptual, modelo lógico, y modelo físico.

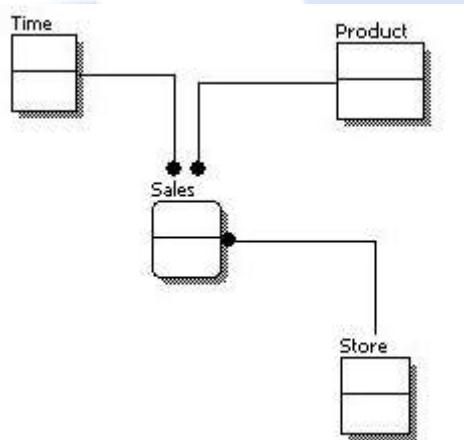
a. Modelos de datos conceptuales

Un modelo conceptual de datos identifica las relaciones de más alto nivel entre las diferentes entidades.

Las características del modelo conceptual de datos incluyen:

- Incluye las entidades importantes y las relaciones entre ellas.
- No se especifica ningún atributo.
- No se especifica ninguna clave principal.

La siguiente figura es un ejemplo de un modelo conceptual de datos.



b. Modelos de datos físicos

El modelo de datos físicos representa cómo se construirá el modelo en la base de datos.

Un modelo de base de datos física muestra todas las estructuras de tabla, incluidos el nombre de columna, el tipo de datos de columna, las restricciones de columna, la clave principal, la clave externa y las relaciones entre las tablas.

Las características de un modelo de datos físicos incluyen:

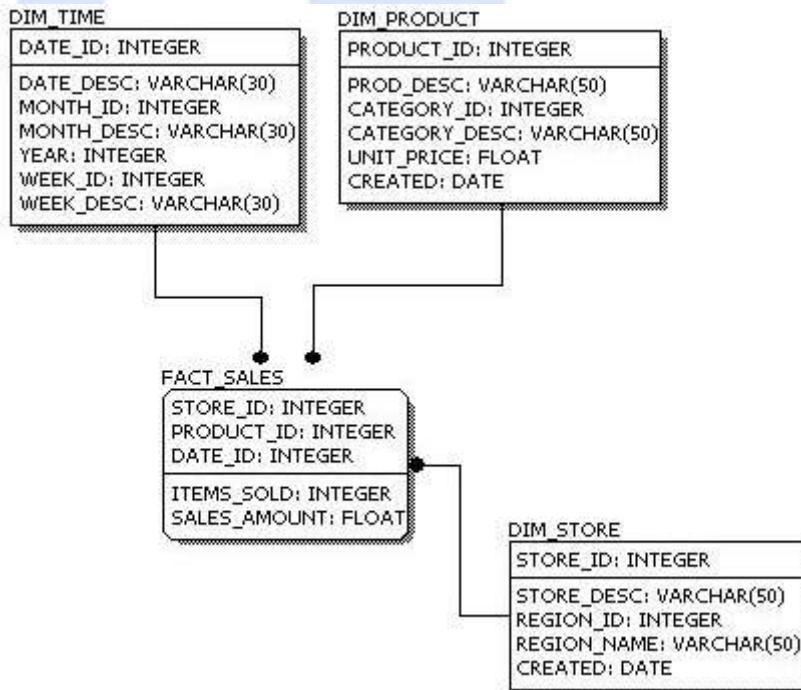
- Especificación de todas las tablas y columnas.
- Las claves externas se usan para identificar relaciones entre tablas.
- La desnormalización puede ocurrir según los requisitos del usuario.

- Las consideraciones físicas pueden hacer que el modelo de datos físicos sea bastante diferente del modelo de datos lógicos.

El modelo de datos físicos será diferente para diferentes **Sistemas de Gestión de Base de datos**. Por ejemplo, el tipo de datos para una columna puede ser diferente entre MySQL y SQL Server.

Los pasos básicos para el diseño del modelo de datos físicos son los siguientes:

- Convertir entidades en tablas.
- Convertir relaciones en claves externas.
- Convertir atributos en columnas.
- Modificar el modelo de datos físicos en función de las restricciones / requisitos físicos.



c. Modelos de datos lógicos

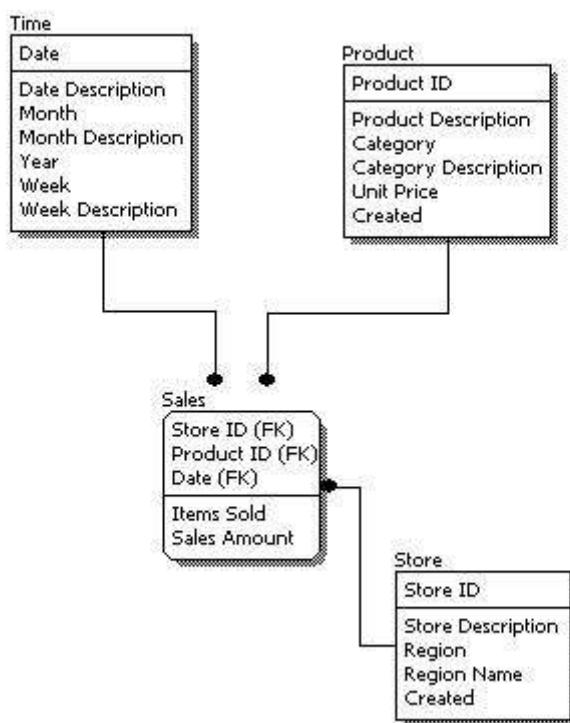
Un modelo de datos lógicos describe los datos con el mayor detalle posible, independientemente de cómo se implementarán físicamente en la base de datos.

Las características de un modelo de datos lógicos incluyen:

- Incluye todas las entidades y relaciones entre ellos.
- Todos los atributos para cada entidad están especificados.

- La clave principal para cada entidad está especificada.
- Se especifican las claves externas (claves que identifican la relación entre diferentes entidades).
- La normalización ocurre en este nivel.
- Los pasos para diseñar el modelo de datos lógicos son los siguientes:
- Especifique claves primarias para todas las entidades.
- Encuentra las relaciones entre diferentes entidades.
- Encuentra todos los atributos para cada entidad.
- Resuelva las relaciones de muchos a muchos.
- Normalización.

La siguiente figura es un ejemplo de un modelo de datos lógicos.



Aquí comparamos estos tres tipos de modelos de datos. La tabla a continuación compara las diferentes características:

Característica	Conceptual	Lógico	Física
Nombres de entidades	✓	✓	
Relaciones de entidades	✓	✓	
Atributos		✓	
Primary Keys		✓	✓
Foreign Keys		✓	✓
Nombres de tabla			✓
Nombres de columnas			✓
Tipos de datos de columna			✓

Ventajas y desventajas de los modelos de datos

Ventajas

- El objetivo principal de un modelo de datos es asegurarse de que los objetos de datos ofrecidos por el equipo funcional se representen con precisión.
- El modelo de datos debe ser lo suficientemente detallado para ser utilizado para construir la base de datos física.
- La información en el modelo de datos se puede utilizar para definir la relación entre tablas, claves primarias y externas y procedimientos almacenados.
- El modelo de datos ayuda a las empresas a comunicarse dentro y entre las organizaciones.
- El modelo de datos ayuda a documentar las asignaciones de datos en el proceso ETL
- Ayuda a reconocer las fuentes de datos correctas para poblar el modelo.

Desventajas

- Para desarrollar el modelo de datos se deben conocer las características físicas de los datos almacenados.
- Incluso los cambios más pequeños realizados en la estructura requieren modificaciones en toda la aplicación.
- No hay un lenguaje de manipulación de modelos establecido en DBMS.

Bibliografía:

<https://bbddrafaelcampanero.wordpress.com/2016/09/29/evolucion-de-los-sistemas-de-gestion-de-bases-de-datos/>



