



Modelo Conceptual de Datos II

Bases de datos I – UTN FRRO

Giuliano Crenna



Modelo de Datos: Definición y Características

Un **modelo de datos** es un sistema formal y abstracto que permite describir los datos de acuerdo con reglas y convenios predefinidos. Este concepto es fundamental en el campo de la gestión de la información, ya que proporciona una estructura organizada y coherente para representar los datos, facilitando su análisis y manipulación.

Formalidad del Modelo de Datos

El modelo de datos se considera formal porque los objetos dentro del sistema se describen siguiendo reglas y operadores claramente definidos. Esta formalidad asegura que las representaciones de los datos sean consistentes y precisas, independientemente del significado que esos objetos y operadores puedan tener en el contexto del mundo real. En otras palabras, la formalidad del modelo garantiza que la interpretación de los datos sea uniforme y no esté sujeta a ambigüedades.

Abstracción en el Modelo de Datos

El modelo de datos es también abstracto, lo que significa que se obtiene a través de un proceso de abstracción. Este proceso implica seleccionar y representar solo aquellos aspectos del mundo real que son relevantes para los objetivos del sistema o del dominio de estudio. La abstracción permite simplificar la complejidad del mundo real, enfocándose en los elementos más significativos para la aplicación en cuestión. Así, se omiten los detalles irrelevantes, facilitando un enfoque más directo y eficaz en la gestión de los datos.

Componentes de un Modelo de Datos

Un **modelo de datos** se compone de tres elementos fundamentales que, en conjunto, permiten la representación, manipulación y validación de la información dentro de un sistema. Estos componentes son: las estructuras de datos, los operadores o reglas de inferencia, y las reglas de integridad.

1. Colección de Estructuras de Datos

Las estructuras de datos son el primer componente clave de un modelo de datos. Se refieren a las formas organizativas mediante las cuales se almacenan y representan los datos en un sistema. Estas estructuras pueden variar ampliamente en complejidad, desde listas y tablas simples hasta estructuras más avanzadas como árboles o grafos. La organización eficiente de los datos es crucial para facilitar su acceso y manipulación.

2. Colección de Operadores o Reglas de Inferencia

El segundo componente incluye los operadores o reglas de inferencia, que son las herramientas utilizadas para consultar o derivar datos a partir de las estructuras existentes. Estos operadores permiten combinar y manipular los datos en cualquier forma deseada, permitiendo extraer información valiosa y generar nuevos conocimientos a partir de los datos almacenados. La flexibilidad y capacidad de estas reglas son esenciales para la versatilidad del modelo de datos.

3. Colección de Reglas de Integridad

El tercer componente son las reglas de integridad, que definen un conjunto de estados consistentes dentro del sistema. Estas reglas garantizan que los datos dentro del modelo sean válidos y coherentes, evitando errores y redundancias. Pueden ser expresadas explícita o implícitamente y suelen incluir directrices para operaciones comunes como insertar, actualizar o borrar datos. Las reglas de integridad son fundamentales para mantener la calidad y confiabilidad de la información en el sistema.

Categorías de los Modelos de Datos

Los modelos de datos se pueden clasificar en diferentes categorías según su nivel de abstracción y cercanía con la representación real de los datos. Estas categorías se dividen en modelos de datos de alto nivel o conceptuales, y modelos de datos de bajo nivel o físicos. A continuación, se describen las principales características de cada una.

Modelos de Datos de Alto Nivel o Conceptuales

Los modelos de datos de alto nivel, también conocidos como modelos conceptuales, están diseñados para ser comprensibles y manejables por la mayoría de los usuarios, ya que se asemejan a la manera en que los usuarios perciben los datos. Se subdividen en dos tipos principales: basados en objetos y basados en registros.

- **Modelos Basados en Objetos:**
 - **Modelo de Dominio:** Este modelo se centra en representar los conceptos y relaciones que son relevantes dentro de un dominio específico, permitiendo una visión clara y directa de los elementos clave que componen ese entorno.
 - **Modelo Entidad-Relación (ER):** Utilizado para representar datos y sus relaciones mediante entidades (objetos) y las relaciones entre ellas. Es comúnmente utilizado en el diseño de bases de datos.
- **Modelos Basados en Registros:**
 - **Modelo Jerárquico:** En este modelo, los datos se organizan en registros que están relacionados entre sí mediante punteros, y se estructuran como colecciones de árboles. Cada registro tiene un único padre, lo que crea una jerarquía de datos.
 - **Modelo de Redes:** Similar al modelo jerárquico, pero con mayor flexibilidad, ya que permite que un registro tenga múltiples relaciones, organizándose como un grafo. Esto permite representar relaciones más complejas entre los datos.
 - **Modelo Relacional:** Organiza los datos en tablas, donde las relaciones entre los datos se establecen mediante el contenido de ciertas columnas, llamadas claves. Es el modelo más utilizado en la actualidad debido a su simplicidad y flexibilidad.

Modelos de Datos de Bajo Nivel o Físicos

Por otro lado, los modelos de datos de bajo nivel o físicos se centran en describir cómo los datos se almacenan y se gestionan dentro de la computadora. Estos modelos proporcionan detalles técnicos sobre el almacenamiento de los datos, como las estructuras de almacenamiento, el acceso a los datos, y los métodos de recuperación. A diferencia de los modelos conceptuales, estos modelos son más abstractos y se utilizan principalmente por ingenieros de sistemas y administradores de bases de datos.

Modelo Entidad-Relación (ER) y Diagrama Entidad-Relación (DER)

El **Modelo Entidad-Relación (ER)** es una metodología fundamental en el diseño de bases de datos, publicada por Peter Chen en 1976. Este modelo se destaca por su carácter comprehensivo, ya que permite representar los datos de manera abstracta y clara, sin preocuparse por las complicaciones relacionadas con el almacenamiento o la eficiencia. Estas consideraciones son reservadas para el diseño físico de la base de datos, lo que permite a los diseñadores centrarse primero en la estructura conceptual de la información.

Evolución del Modelo ER

Con el tiempo, algunos autores han extendido el Modelo ER para incrementar sus capacidades, dando lugar al Modelo ER Extendido (EER). Este modelo avanzado permite representar estructuras de datos más complejas y abordar necesidades específicas que el modelo original no cubría completamente. Estas extensiones incluyen nuevas formas de representar jerarquías, relaciones especiales y atributos adicionales que enriquecen el proceso de modelado.

Diagrama Entidad-Relación (DER)

El enfoque principal del Modelo ER es simplificar el proceso de modelado de datos mediante el desarrollo de un **Diagrama Entidad-Relación (DER)**. Este diagrama es una representación gráfica y abstracta del dominio analizado, en el cual se identifican las entidades (objetos o conceptos relevantes) y las relaciones entre ellas. El DER es una herramienta poderosa que facilita la comprensión y comunicación del diseño de la base de datos tanto para desarrolladores como para otros interesados en el proyecto.

Implementación del DER en Sistemas de Gestión de Bases de Datos

Una vez que se ha desarrollado un DER, este puede ser trasladado y adaptado a modelos más apropiados para su implementación en un Sistema de Gestión de Base de Datos (SGBD o DBMS, por sus siglas en inglés). El DER actúa como un puente entre el diseño conceptual y la implementación física de la base de datos, asegurando que la estructura de la información se mantenga coherente y alineada con los objetivos del sistema durante todo el proceso de desarrollo.

Componentes del Diagrama Entidad-Relación (DER)

El **Diagrama Entidad-Relación (DER)** es una herramienta visual utilizada para modelar la estructura de una base de datos, representando las entidades, atributos y relaciones que existen dentro del sistema. Uno de los componentes fundamentales del DER es la **Entidad**, que se refiere a un tipo de información relevante para el sistema y que tiene una existencia independiente en el mundo real.

Entidad en el DER

Una **Entidad** en el contexto de un DER es una “cosa”, “objeto” o “concepto” del mundo real que es de interés para el sistema que se está modelando. Esta entidad puede tener una existencia física o conceptual, dependiendo de su naturaleza y del dominio del sistema.

- **Entidad Física:** Se refiere a objetos tangibles y materiales que existen en el mundo real y que son relevantes para el sistema. Ejemplos de entidades físicas incluyen herramientas, alumnos, máquinas, etc.
- **Entidad Conceptual:** Se refiere a conceptos abstractos que no tienen una representación física, pero que son igualmente importantes para el sistema. Ejemplos de entidades conceptuales incluyen carreras universitarias, puestos de trabajo, roles, etc.

Cada entidad dentro del DER incluye todas las instancias de ese objeto o concepto, es decir, todas las ocurrencias específicas que pertenecen a esa categoría. En un DER, las entidades se representan mediante un rectángulo con el nombre de la entidad inscrito en el medio, lo que proporciona una representación clara y estandarizada de los elementos clave del sistema.

Este componente es esencial en el proceso de modelado de datos, ya que define los elementos fundamentales que serán gestionados y manipulados dentro del sistema de base de datos, y sirve como base para identificar las relaciones y atributos asociados con cada entidad.

Componentes del Diagrama Entidad-Relación (DER): Relación

En el **Diagrama Entidad-Relación (DER)**, la **Relación** es un componente clave que representa la correspondencia, conexión o vínculo entre dos o más entidades, o incluso entre una entidad y sí misma. Las relaciones son fundamentales para comprender cómo interactúan y se asocian las diferentes entidades dentro del sistema modelado.

Definición de Relación

Una **Relación** en el DER describe la manera en que dos o más entidades están conectadas o interactúan entre sí. Estas relaciones pueden reflejar diferentes tipos de vínculos, como asociaciones directas, dependencias, o cualquier otro tipo de interacciones relevantes para el sistema. Por ejemplo, en una base de datos de ventas, una relación podría describir cómo un usuario realiza pedidos de productos, vinculando la entidad "Usuario" con la entidad "Producto".

- **Relación entre Entidades:** Esto se refiere a una conexión entre dos o más entidades distintas. Por ejemplo, un usuario solicita productos, lo que vincula la entidad "Usuario" con la entidad "Producto".
- **Relación con una Misma Entidad:** Una relación también puede vincular una entidad consigo misma, por ejemplo, un empleado supervisa a otro empleado dentro de la misma entidad "Empleado".

Representación Gráfica de la Relación en el DER

En un DER, una relación se representa mediante un rombo con el nombre de la relación inscrito en el medio. Este rombo se conecta a las entidades que vincula a través de líneas, que representan las conexiones entre las entidades y la relación. La claridad de esta representación gráfica facilita la comprensión de cómo las entidades del sistema están interrelacionadas.

Por ejemplo, en la relación "Un usuario solicita sus productos", el rombo con el nombre "solicita" se conectaría a las entidades "Usuario" y "Producto", indicando que existe una relación de solicitud entre estas dos entidades.

Las relaciones son esenciales en el modelado de datos, ya que definen las interacciones clave dentro del sistema y ayudan a estructurar cómo se gestionarán y accederán los datos en la base de datos.

Tipos de Atributos en el Diagrama Entidad-Relación (DER): Simples vs Compuestos

En el **Diagrama Entidad-Relación (DER)**, los atributos son propiedades o características que describen las entidades o relaciones dentro del sistema modelado. Los atributos se clasifican en diferentes tipos según su complejidad, siendo los **atributos simples** y **atributos compuestos** dos de las categorías principales.

Atributos Simples o Atómicos

Un **atributo simple**, también conocido como **atributo atómico**, es aquel que admite un valor único que no puede subdividirse en componentes más pequeños. Estos atributos son indivisibles y representan una unidad básica de información en el sistema.

- **Ejemplos de Atributos Simples:**
 - nombre_herramienta: Representa el nombre de una herramienta y no se puede dividir en partes más pequeñas.
 - fecha_entrega: Indica la fecha y hora de entrega de un ítem, tratándose como una única entidad.
 - nro_legajo: Es un número de identificación que no se subdivide en otras partes.

Los atributos simples son utilizados cuando la información no requiere un desglose adicional y se maneja de manera directa en el sistema.

Atributos Compuestos

Un **atributo compuesto** es aquel que puede dividirse en componentes más pequeños, cada uno de los cuales tiene un significado y puede ser gestionado de manera independiente. Estos atributos se utilizan cuando es necesario descomponer la información en partes más detalladas para un mejor análisis o gestión.

- **Ejemplo de Atributo Compuesto:**
 - contacto_emergencia: Este atributo se puede subdividir en dos componentes más pequeños, nom_cto_emergencia (nombre del contacto de emergencia) y tel_cto_emergencia (teléfono del contacto de emergencia). Cada uno de estos subatributos representa una parte específica de la información contenida en el atributo compuesto.

Los atributos compuestos son útiles en situaciones donde la información tiene múltiples facetas que deben ser consideradas por separado para un manejo más detallado.

Tipos de Atributos en el Diagrama Entidad-Relación (DER): Monovaluados vs Multivaluados

En el **Diagrama Entidad-Relación (DER)**, los atributos pueden clasificarse según el número de valores que pueden almacenar para una instancia de una entidad. Las dos categorías principales en esta clasificación son **atributos monovaluados** y **atributos multivaluados**.

Atributos Monovaluados

Un **atributo monovaluado** es aquel que admite un único valor para cada instancia de una entidad. Estos atributos son comunes cuando cada entidad tiene una sola característica relevante en el contexto de ese atributo.

- **Ejemplo de Atributo Simple y Monovaluado:**
 - nombre_herramienta: Este atributo es simple porque no puede subdividirse y monovaluado porque cada herramienta tiene un único nombre.
- **Ejemplo de Atributo Compuesto y Monovaluado:**
 - contacto_emergencia: Este es un atributo compuesto porque puede dividirse en subatributos como nom_cto_emergencia y tel_cto_emergencia, pero es monovaluado porque para cada instancia de entidad se almacena un solo valor de contacto de emergencia.

Los atributos monovaluados son útiles cuando se requiere que cada entidad tenga una característica única y no múltiple en un determinado contexto.

Atributos Multivaluados

Un **atributo multivaluado** es aquel que puede almacenar varios valores para una única instancia de una entidad. Estos atributos son necesarios cuando es posible que una entidad tenga múltiples instancias de una característica particular.

- **Ejemplo de Atributo Simple y Multivaluado:**
 - teléfonos: Este atributo es simple porque no puede subdividirse en partes más pequeñas, pero es multivaluado porque un operario puede tener uno o varios números de teléfono asociados. Además, un atributo multivaluado puede tener un límite en la cantidad de valores que puede almacenar. Por ejemplo, un operario podría tener al menos un teléfono y como máximo dos.

Los atributos multivaluados son cruciales cuando es necesario capturar y gestionar múltiples valores relacionados con una sola instancia de entidad, permitiendo una representación más completa y flexible de la información.

Identificadores o Claves en el Diagrama Entidad-Relación (DER)

En el **Diagrama Entidad-Relación (DER)**, los **identificadores o claves** son atributos fundamentales que permiten identificar de manera única cada instancia de una entidad. Estos identificadores aseguran que cada registro dentro de una base de datos sea único y distinguible de los demás, lo que es crucial para la integridad y consistencia de los datos.

Definición de Identificadores o Claves

Cada entidad en un DER debe poseer un atributo identificador, o un conjunto de ellos cuando la clave es compuesta, que permita diferenciar unívocamente cada una de sus instancias. Este atributo o conjunto de atributos es esencial para realizar operaciones precisas sobre los datos, como búsqueda, actualización o eliminación de registros.

Características de los Identificadores o Claves

Los atributos que conforman un identificador o clave deben cumplir con dos características esenciales:

1. **Monovaluados:** El atributo identificador debe ser monovaluado, es decir, debe admitir un único valor para cada instancia de la entidad. Esto asegura que cada registro sea asociado con una única clave.
2. **No Nulos (NOT NULL):** El atributo o conjunto de atributos que conforman la clave debe siempre tener un valor asignado, lo que significa que no puede ser nulo. Esto garantiza que cada entidad tenga un identificador único y válido en todo momento.

Ejemplos de Identificadores o Claves

- **ID:** En una entidad que representa herramientas, el atributo ID podría ser utilizado como identificador, ya que identifica de manera única cada herramienta en la base de datos.
- **NRO_LEGajo:** En una entidad que representa operarios, el atributo NRO_LEGajo podría ser utilizado como identificador, permitiendo identificar de manera única a cada operario.

Los identificadores o claves son fundamentales en el diseño de bases de datos porque proporcionan la base para establecer relaciones entre diferentes entidades, realizar consultas eficientes, y mantener la integridad de los datos. La correcta definición y uso de identificadores es esencial para un sistema de información robusto y bien estructurado.

Restricciones en el Diagrama Entidad-Relación (DER)

En el **Diagrama Entidad-Relación (DER)**, las **restricciones** son fundamentales para definir las reglas que regulan las relaciones entre entidades y asegurar la integridad de la base de datos. Estas restricciones incluyen la **razón de cardinalidad** y la **restricción de participación**, cada una con su propia importancia y representación en el modelo.

Razón de Cardinalidad

La **razón de cardinalidad** se refiere a la cantidad de instancias de una entidad que pueden participar en una relación con instancias de otra entidad. Define el número mínimo y máximo de instancias que una entidad puede tener en una relación específica.

- **Representación de la Cardinalidad:** Aunque no existe un estándar universal para representar la cardinalidad en una relación, se utilizan etiquetas con cardinalidad mínima y máxima para describirla. Las notaciones comunes incluyen:
 - 0..1: La entidad puede participar en la relación de 0 a 1 vez.
 - 1..1: La entidad debe participar en la relación exactamente 1 vez.
 - 0..M: La entidad puede participar en la relación de 0 a muchos veces.
 - 1..M: La entidad debe participar en la relación al menos 1 vez y hasta muchos veces.
 - M..N: La entidad puede participar en la relación entre M y N veces.
- **Consideración de Cardinalidad:** En la representación, la cardinalidad del vínculo se menciona siempre por sus valores máximos, especificando el límite superior de la participación de la entidad en la relación.

Restricción de Participación

La **restricción de participación** define si la existencia de una instancia de una entidad depende de su relación con otra entidad. Esta restricción se clasifica en dos tipos principales:

- **Total (Dependencia de Existencia):** Una entidad tiene una dependencia total si su existencia está completamente condicionada a la existencia de una instancia de otra entidad. Es decir, la entidad no puede existir sin estar relacionada con la otra.
- **Parcial:** Una entidad tiene una dependencia parcial si su existencia no depende completamente de la relación con otra entidad. En este caso, la entidad puede existir independientemente de su relación con la otra.

Estas restricciones son esenciales para modelar correctamente las interacciones entre entidades y garantizar que las reglas de negocio y las dependencias se reflejen adecuadamente en el diseño de la base de datos.



Bibliografía:

- **Elmasri, R., & Navathe, S. B.** (2015). *Fundamentals of Database Systems* (7th ed.). Pearson.
- **Chen, P. P.** (1976). *The Entity-Relationship Model—Toward a Unified View of Data*. ACM Transactions on Database Systems, 1(1), 9-36.
- **Harrington, J. L.** (2016). *Relational Database Design and Implementation* (3rd ed.). Morgan Kaufmann.
- **Korth, H. F., & Silberschatz, A.** (2010). *Database System Concepts* (6th ed.). McGraw-Hill.