

FSID

*Fundamentos de los Sistemas de Información
Digitales*

Introducción al Diseño de bases de datos Relacionales

Año 2025



Bases de Datos Relacionales

Las tablas de bases de datos relacionales se pueden combinar fácilmente para ofrecer los datos requeridos por los usuarios, siempre y cuando dos tablas compartan un elemento de datos común.

En una base de datos relacional se utilizan tres operaciones básicas, para desarrollar conjuntos útiles de datos: seleccionar, unir y proyectar.

La operación **seleccionar** crea un subconjunto que consiste en todos los registros del archivo que cumplan con criterios establecidos. En otras palabras, la selección crea un subconjunto de filas que cumplen con ciertos criterios.

La operación **unir** combina tablas relacionales para proveer al usuario con más información de la que está disponible en una tabla individual.

La operación **proyectar** crea un subconjunto que consiste en columnas en una tabla, con lo cual el usuario puede crear nuevas tablas que contengan solamente la información requerida.

Diseño de bases de Datos

Los sistemas de bases de datos se diseñan para gestionar grandes cantidades de información.

El diseño de bases de datos implica principalmente el diseño del esquema de las bases de datos.

El diseño de un entorno completo de aplicaciones para la base de datos que satisfaga las necesidades de la empresa que se está modelando exige prestar atención a un conjunto de aspectos más amplio.

Nos centraremos inicialmente en la escritura de las consultas a la base de datos y en el diseño de los esquemas de las bases de datos.

Proceso de diseño

Lo primero que se debe conocer son los requisitos de datos de los usuarios de las bases de datos y la manera en que se estructurará la base de datos para satisfacer esos requisitos.

La fase inicial del diseño de las bases de datos es caracterizar completamente los requisitos de datos de los hipotéticos usuarios de la base de datos.

El resultado de esta fase es la especificación de los requisitos de los usuarios.

El diseñador escoge un modelo de datos y, mediante la aplicación de los conceptos del modelo de datos elegido, traduce esos requisitos en un esquema conceptual de la base de datos. El esquema desarrollado en esta fase de **diseño conceptual** ofrece una visión general detallada de la empresa.

El diseñador revisa el esquema para confirmar que todos los requisitos de datos se satisfacen realmente y no entren en conflicto entre sí.

Proceso de Diseño (Continuación)

El diseñador debe examinar el diseño para eliminar cualquier característica redundante.

En este punto, la atención se centra en describir los datos y sus relaciones, más que en especificar los detalles del almacenamiento físico.

En términos del modelo relacional, el proceso de diseño conceptual implica decisiones sobre qué atributos se desea capturar en la base de datos y cómo agruparlos para formar las diferentes tablas.

La parte “que” es una decisión conceptual.,

La parte del “cómo” es, esencialmente, un problema informático.

Proceso de Diseño (Continuación)

Hay dos formas para resolver el problema.

La primera supone usar el modelo entidad-relación ;

la otra es emplear un conjunto de algoritmos (denominados colectivamente como normalización) que toma como entrada el conjunto de todos los atributos y genera un conjunto de tablas.

Proceso de diseño (continuación)

Un esquema conceptual completamente desarrollado también indica los requisitos funcionales de la empresa. En la especificación de requisitos funcionales los usuarios describen el tipo de operaciones (o transacciones) que se llevarán a cabo con los datos. Un ejemplo de estas operaciones es modificar o actualizar los datos, buscar y recuperar datos concretos y eliminar datos.

En esta etapa del diseño conceptual el diseñador puede revisar el esquema para asegurarse de que satisface los requisitos funcionales.

El proceso de pasar de un modelo de datos abstracto a la implementación de la base de datos continúa con dos fases de diseño finales. En la fase de diseño lógico el diseñador relaciona el esquema conceptual de alto nivel con el modelo de implementación de datos del sistema de bases de datos que se va a usar.

El diseñador usa el esquema de bases de datos específico para el sistema resultante en la fase de diseño físico posterior, en la que se especifican las características físicas de la base de datos. Entre esas características están la forma de organización de los archivos y las estructuras de almacenamiento interno.

Modelo entidad-Relación (E-R)

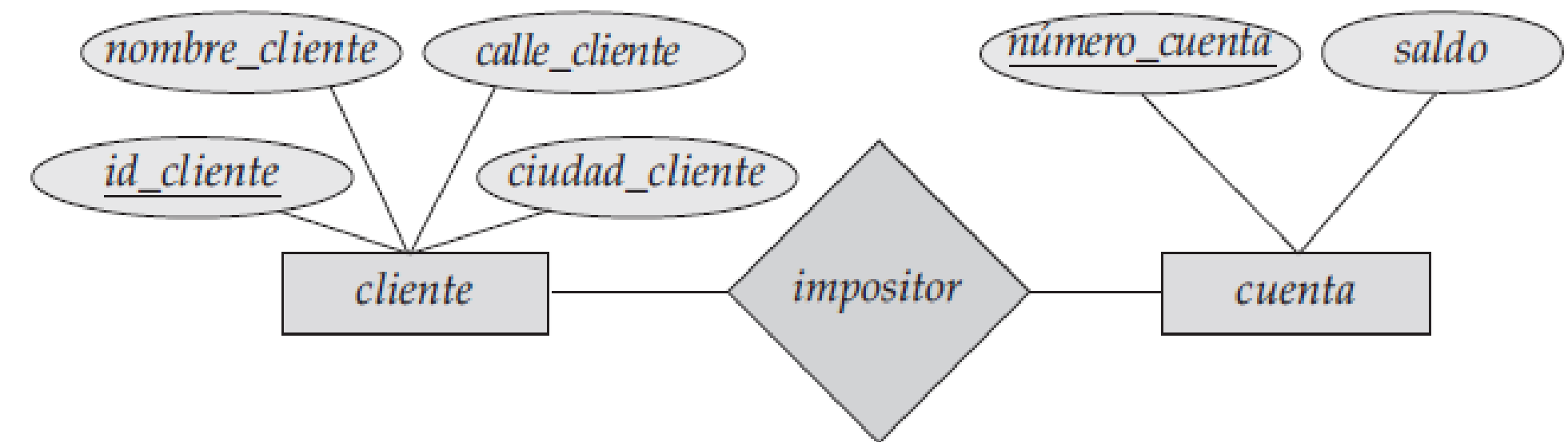
Basado en una percepción del mundo real que consiste en un conjunto de objetos básicos, denominados **entidades**, y de las **relaciones** entre esos objetos.

Una **entidad** es una cosa u objeto del mundo real que es distinguible de otros objetos.

Las **entidades** se describen en las bases de datos mediante un conjunto de atributos.

Se usa un atributo extra para identificar unívocamente a cada ocurrencia de una entidad.

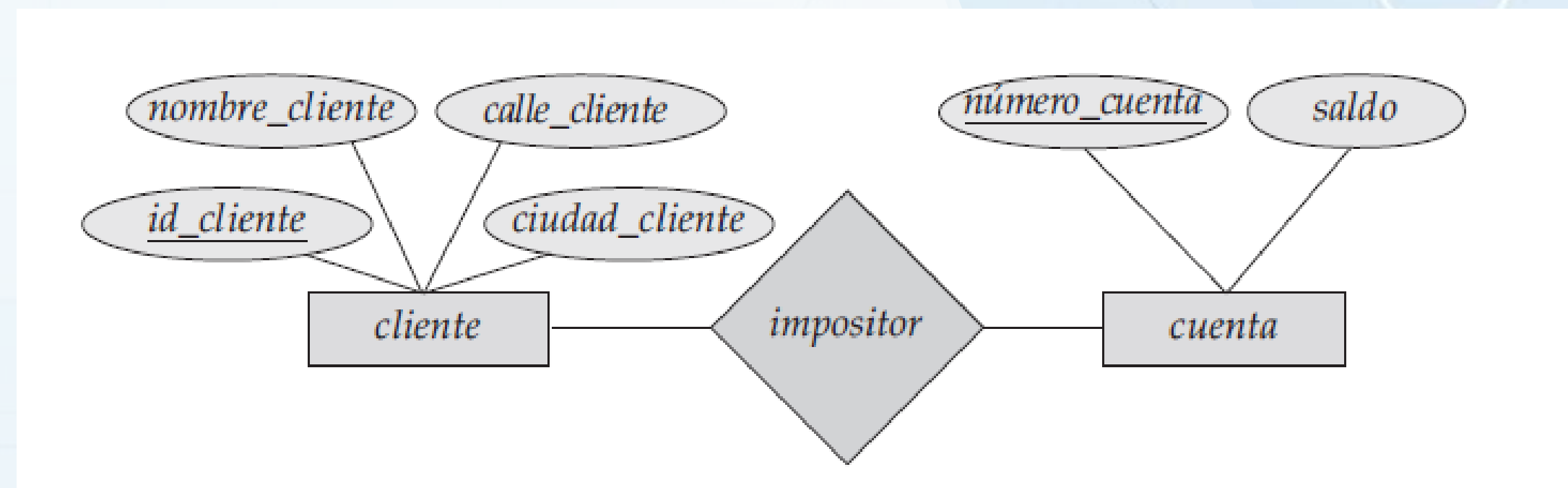
Una **relación** es una asociación entre varias entidades



Modelo entidad-Relación (E-R)

La estructura lógica general (esquema) de la base de datos se puede expresar gráficamente mediante un diagrama E-R

- Rectángulos: Que representan conjuntos de entidades. Su nombre se pone en singular.
- Elipses: Que representan atributos.
- Rombo: Que representan conjuntos de relaciones entre miembros de varios conjuntos de entidades.
- Líneas: Que unen los atributos con los conjuntos de entidades entre sí, y también los conjuntos de entidades con las relaciones.



Ejemplo entidad bancaria

Para ilustrar el proceso de diseño, examinemos el modo en que puede diseñarse una base de datos para una entidad bancaria. La especificación inicial de los requisitos de los usuarios puede basarse en entrevistas con los usuarios de la base de datos y en el análisis de la entidad realizado por el propio diseñador. La descripción que surge de esta fase de diseño sirve como base para especificar la estructura conceptual de la base de datos.

Principales características de la entidad bancaria:

- El banco está organizado en sucursales. Cada sucursal se ubica en una ciudad determinada y queda identificada por un único nombre. El banco supervisa los activos de cada sucursal.
- Los clientes quedan identificados por el valor de su `id_cliente`. El banco almacena el nombre de cada cliente, la calle y la ciudad en la que vive. Los clientes pueden abrir cuentas y solicitar préstamos. Cada cliente puede asociarse con un empleado del banco en concreto, que puede actuar como prestamista o como asesor personal para ese cliente.
- El banco ofrece dos tipos de cuenta: de ahorro y corriente. Las cuentas pueden tener como titular a más de un cliente y cada cliente puede abrir más de una cuenta. A cada cuenta se le asigna un número de cuenta único. El banco guarda un registro del saldo de cada cuenta y de la fecha más reciente en que cada cliente titular de esa cuenta tuvo acceso a ella. Además cada cuenta de ahorro tiene una tasa de interés y se registran los descubiertos de las cuentas corrientes.

Ejemplo entidad bancaria (continuación)

- El banco ofrece préstamos a sus clientes. Cada préstamo se origina en una sucursal concreta y puede tener como titulares a uno o más clientes. Cada préstamo queda identificado por un número de préstamo único. de cada préstamo el banco realiza un seguimiento del importe del préstamo y de sus pagos. Aunque el número de pago del préstamo no identifica de manera única un pago concreto entre todos los del banco, el número de pago identifica cada pago concreto de un préstamo dado. Se registran la fecha y el importe de cada pago.
- Los empleados del banco quedan identificados por el valor de su id_empleado. La administración del banco almacena el nombre y número de teléfono de cada empleado , el nombre de las personas dependientes de cada empleado y el número de id_empleado del jefe de cada empleado. El banco también realiza un seguimiento de la fecha de contratación y por lo tanto, de la antigüedad de cada empleado.

Normalización

Otro método de diseño de bases de datos es usar un proceso que suele denominarse **normalización**.

El objetivo es generar un conjunto de esquemas de relaciones que permita almacenar información sin redundancias innecesarias, pero que también permita recuperar la información con facilidad. El enfoque es diseñar esquemas que se hallen en la forma normal adecuada.

Para comprender la necesidad de la normalización, obsérvese lo que puede fallar en un mal diseño de base de datos.

Algunas de las propiedades no deseables son:

- Repetición de la información.
- Imposibilidad de representar determinada información.

Mineria y Analisis de datos

El término **minería de datos** se refiere en líneas generales al proceso de análisis semiautomático de grandes bases de datos para descubrir patrones útiles.

Al igual que el descubrimiento de conocimiento en inteligencia artificial (también denominado Aprendizaje de la máquina o machine learning) o el análisis estadístico, la minería de datos intenta descubrir reglas y patrones en los datos.

Sin embargo, la minería de datos se diferencia del aprendizaje de la máquina y de la estadística en que maneja grandes volúmenes de datos, almacenados principalmente en disco. Es decir, la minería de datos trata del descubrimiento de conocimiento en las bases de datos.

Algunos tipos de conocimiento descubierto en las bases de datos puede representarse mediante un conjunto de reglas. Otros tipos de conocimientos se representan mediante ecuaciones que relacionan diferentes variables, o mediante otros mecanismos para la predicción de los resultados cuando se conocen los valores de algunas variables.

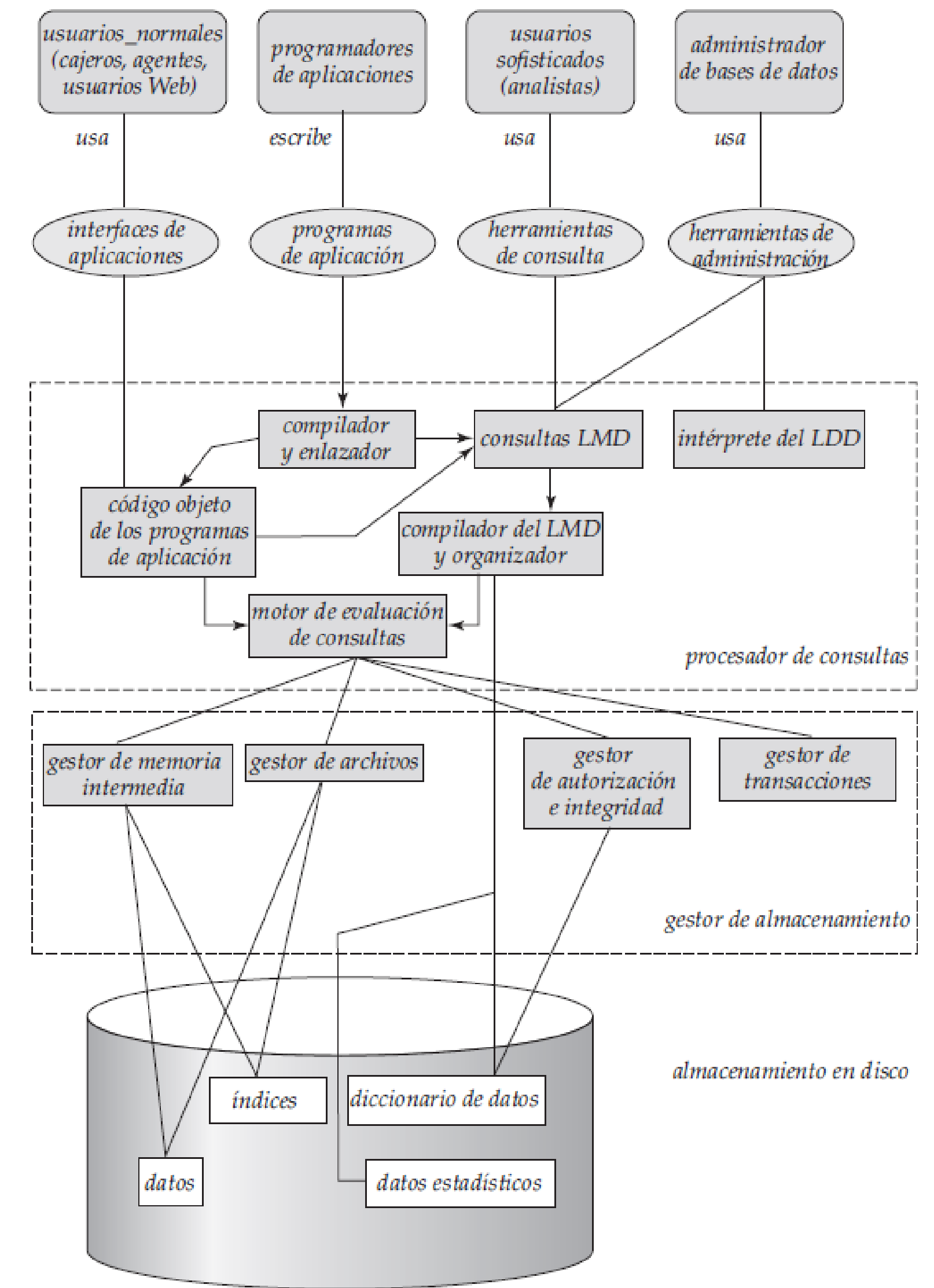
Arquitectura de bases de datos

La arquitectura de los sistemas de bases de datos se ve muy influida por el sistema informático subyacente sobre el que se ejecuta el sistema de bases de datos.

Los sistemas de bases de datos pueden estar centralizados o ser del tipo cliente-servidor. También se pueden diseñar para aprovechar las arquitecturas de computadoras paralelas.

Las bases de datos distribuidas se extienden por varias máquinas geográficamente separadas.

Actualmente la mayor parte de los usuarios de los sistemas de bases de datos no está presente en el lugar físico en que se encuentra el sistema de bases de datos, sino que se conectan a él a través de una red. Por lo tanto, se puede diferenciar entre los sistemas clientes, en los que trabajan los usuarios remotos de la base de datos y los sistemas servidores, en los que se ejecutan los sistemas de bases de datos.



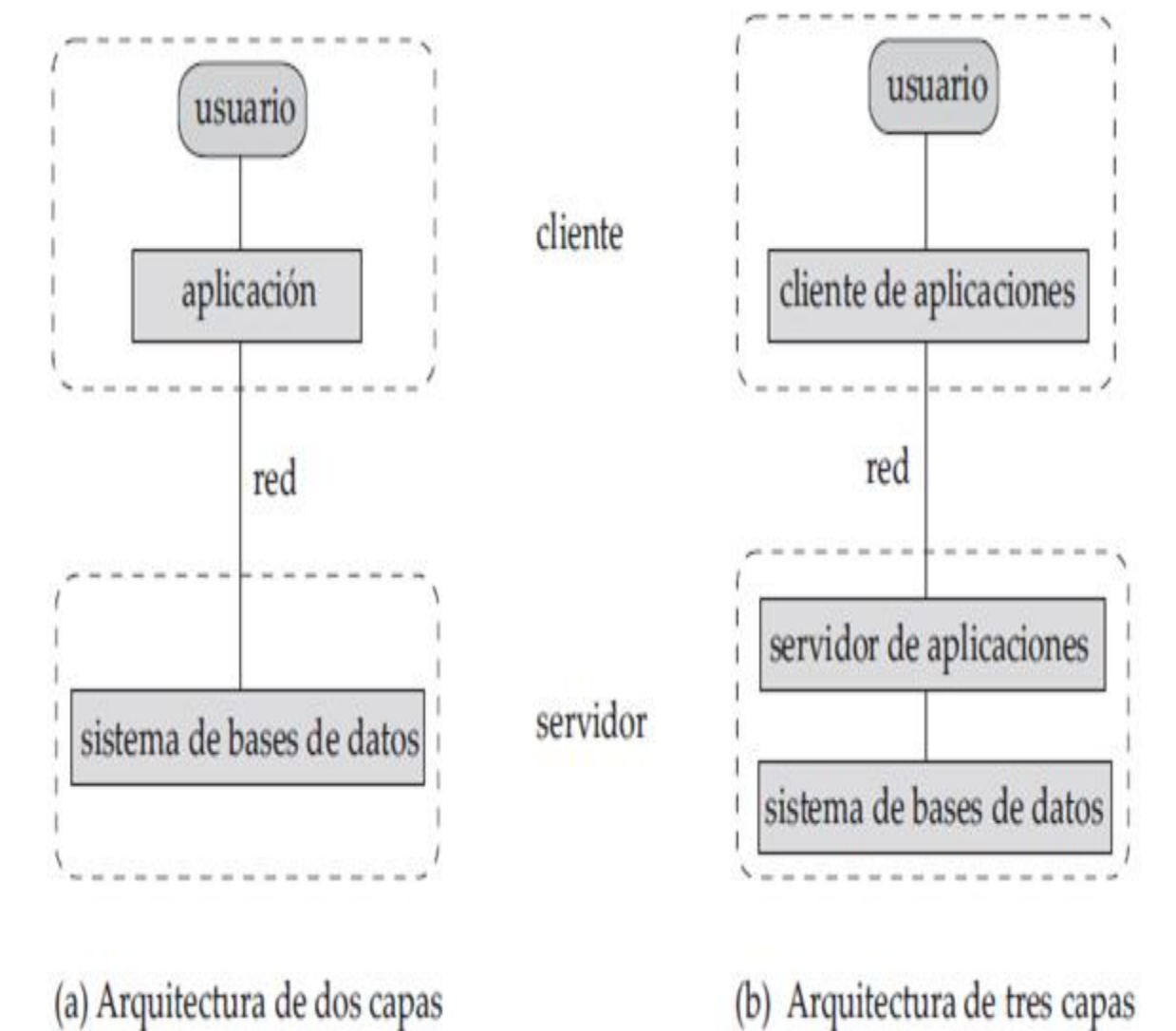
Arquitectura de bases de datos

Las aplicaciones de bases de datos suelen dividirse en dos o tres partes:

La arquitectura de dos capas la aplicación se divide en un componente que reside en la máquina cliente, que llama a la funcionalidad del sistema de bases de datos en la máquina servidora mediante instrucciones del lenguaje de consultas. Los estándares de interfaces de programas de aplicación como ODBC y JDBC se usan entre el cliente y el servidor.

En una arquitectura de tres capas, la máquina cliente actúa simplemente como una parte visible al usuario y no contiene ninguna llamada directa a la base de datos. En vez de eso, el extremo cliente se comunica con un servidor de aplicaciones, generalmente mediante una interfaz de formulario. El servidor de aplicaciones, a su vez, se comunica con el sistema de bases de datos para tener acceso a los datos. La lógica de negocio de la aplicación, que establece las acciones que se deben realizar según las condiciones reinantes, se incorpora en el servidor de aplicaciones, en lugar de estar distribuida entre múltiples clientes.

Las aplicaciones de tres capas resultan más adecuadas para aplicaciones de gran tamaño y para las aplicaciones que se ejecutan en la web.



Usuarios y administradores de bases de datos

Usuarios Normales: Son usuarios no sofisticados que interactúan con el sistema invocando alguno de los programas de aplicación que se han escrito previamente. La interfaz de usuario habitual para los usuarios normales es una interfaz de formularios, donde el usuario puede rellenar los campos correspondientes del formulario. Los usuarios normales también pueden limitarse a leer informes generados por la base de datos.

Programadores de aplicaciones: Son profesionales informáticos que escriben programas de aplicación. Los programadores de aplicaciones pueden elegir entre muchas herramientas para desarrollar las interfaces de usuario. Las herramientas de **desarrollo rápido de aplicaciones** (DRA) son herramientas que permiten al programador de aplicaciones crear formularios e informes con un mínimo esfuerzo de programación.

Usuarios sofisticados: Interactúan con el sistema sin escribir programas, formulan sus consultas en lenguaje de consultas de bases de datos. Remiten cada una de las consultas al procesador de consultas, cuya función es dividir las instrucciones LMD en instrucciones que el gestor de almacenamiento entienda. Los analistas que remiten las consultas para explorar los datos de la base de datos entran en esta categoría.

Usuarios especializados: son usuarios sofisticados que escriben aplicaciones de bases de datos especializadas que no encajan en el marco tradicional del procesamiento de datos. Entre estas aplicaciones están los sistemas de diseño asistido por computadora, los sistemas de bases de conocimientos y los sistemas expertos, los sistemas que almacenan datos con tipos de datos complejos (por ejemplo los datos gráficos o los datos de sonido) y los sistemas de modelado del entorno.

Usuarios y administradores de bases de datos

Administrador de base de datos: Una de las principales razones de usar SGBD (sistemas de gestión de base de datos) es tener un control centralizado tanto de los datos como de los programas que tienen acceso a esos datos. La persona que tiene ese control central sobre el sistema se denomina administrador de base de datos (ABD) y sus funciones son:

Definición del esquema: el ABD crea el esquema original de la BD mediante la ejecución de un conjunto de instrucciones de definición de datos en el LDD.

Definición de la estructura y del método de acceso.

Modificación del esquema y de la organización física: el ABD realiza modificaciones en el esquema y en la organización física para reflejar las necesidades cambiantes de la organización o alterar la organización física a fin de mejorar el rendimiento.

Concesión de autorización para el acceso de los datos: el ABD puede regular las partes de la base de datos a las que puede tener acceso cada usuario.

Mantenimiento rutinario: Copias de seguridad periódicas de las BD, asegurarse que se dispone de suficiente espacio en disco, supervisar los trabajos que se ejecuten en la base de datos y asegurarse que el rendimiento no se degrade.

Claves

Es necesario disponer de un modo de especificar la manera en que las tuplas de una relación dada se distingan entre sí. Esto se expresa en términos de sus atributos. Es decir, los valores de los valores de los atributos de una tupla deben ser tales que puedan identificarla unívocamente.

Una **superclave** es un conjunto de uno o varios atributos que, considerados conjuntamente, permiten identificar de manera unívoca una tupla de la relación.

Las superclaves que estén expresadas en su mínima expresión y no tengan atributos innecesarios se denominan **claves candidatas**.

Se usará el término **clave primaria** para denotar una clave candidata que ha elegido el diseñador de la base de datos como medio principal para la identificación de las tuplas de una relación.

La clave primaria debe escogerse de manera que los valores de sus atributos no se modifiquen nunca, o muy rara vez.

El esquema de una relación puede incluir entre sus atributos la clave primaria de otro esquema de relación, por ejemplo r_2 . Este atributo se denomina **clave externa o foránea**

Lenguajes de consultas

Un lenguaje de consultas es un lenguaje en el que los usuarios solicitan información de la base de datos. Estos lenguajes suelen ser de un nivel superior que el de los lenguajes de programación habituales. Los lenguajes de consultas pueden clasificarse como procedimentales o no procedimentales.

En los lenguajes **procedimentales** el usuario indica al sistema que lleve a cabo una serie de operaciones en la base de datos para calcular el resultado deseado.

En los lenguajes **no procedimentales** el usuario describe la información deseada sin dar un procedimiento concreto para obtener esa información.

La mayor parte de los sistemas comerciales de bases de datos relacionales ofrecen un lenguaje de consultas que incluye elementos de los enfoques procedimental y no procedimental. Nos referimos al muy usado lenguaje de consultas SQL

Glosario de términos

Un **bit** representa la unidad más pequeña de datos que una computadora puede manejar.

Un grupo de 8 bits se denominado **byte**, representa a un solo carácter, que puede ser una letra, un número u otro símbolo.

Un conjunto de palabras o un número completo (como el nombre o la edad de una persona) se denomina **campo**.

Un grupo de campos relacionados, como el nombre del estudiante, el curso que va a tomar, la fecha y la calificación, representan un **registro**.

Un grupo de registros del mismo tipo se denomina **Tabla**.

Un grupo de Tablas relacionados constituye una base de datos.

Un registro describe a una entidad. Una entidad es una persona, lugar, cosa o suceso sobre el cual almacenamos y mantenemos información.

Cada característica o cualidad que describe a una entidad específica se denomina atributo. Por ejemplo, ID_Estudiante, Curso, Fecha y Calificaciones.

FSID

GRACIAS

Año 2025

