

BASE DE DATOS



ARQUITECTURA DE TRES NIVELES

El objetivo de la arquitectura de tres niveles es la separación entre las aplicaciones de usuario y la base de datos física.

Los tres esquemas (que corresponden a los tres niveles) son sólo una forma de descripción de los datos. Los únicos datos que existen están en el nivel físico.



ARQUITECTURA DE UNA BASE DE DATOS

Nivel Interno : Más cercano al almacenamiento físico, se ocupa de cómo se almacenan físicamente los datos.

Nivel Conceptual : Es el nivel intermedio entre los otros dos y presenta una vista global o comunitaria de los datos.

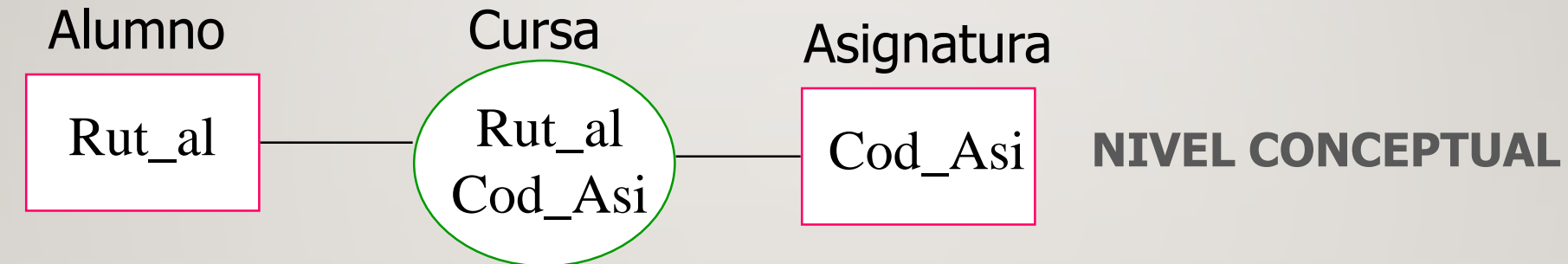
Nivel Externo Más cercano a los usuarios, se ocupa de como los usuarios perciben los datos (vistas individuales).



Arquitectura de tres niveles (Ejemplo)

“ Listado de alumnos por Asignatura
Asignaturas de un Alumno ”

NIVEL EXTERNO



Rut_al N7
Dig_V C1
Nom_Al C24

.....

Rut_al N7
Cod_Asi N6
Sec N1

Cod_Asi N6
Nom_As C16
Num_Cr N2

.....

NIVEL INTERNO

INDEPENDENCIA LÓGICA Y FÍSICA

El esquema conceptual debe ser absolutamente independiente del físico. Esto significa:

- Independencia física de los datos. Aunque el esquema físico cambie, el esquema conceptual no debe verse afectado. En la práctica esto significa que aunque se añadan o cambien discos u otro hardware, o se modifique el sistema operativo u otros cambios relacionados con la física de la base de datos, el esquema conceptual permanece invariable.
- Independencia lógica de los datos. Significa que aunque se modifique el esquema conceptual, la vista que poseen las aplicaciones (los esquemas externos) no serán afectados.

MODELOS DE BASE DE DATOS



MODELOS DE DATOS

Su finalidad es la de simbolizar una parte del mundo real de forma que sea más fácilmente manipulable.

Un modelo de datos para las bases de datos es una colección de conceptos que se emplean para describir la estructura de una base de datos. Esa colección de conceptos incluyen: entidades, atributos y relaciones. Las bases de datos almacenan datos, permitiendo manipularlos fácilmente y mostrarlos de diversas formas. El proceso de construir una base de datos es llamado diseño de base de datos.

Hace énfasis en cómo los datos están representados en la base de datos o en cómo se ejecutan las estructuras de datos para representar lo que está modelado.



MODELO JERÁRQUICO

Organiza los datos según una estructura de árbol invertido en el cual cada registro se conoce como segmento.

Cada segmento guarda una relación 1:M con el segmento directamente debajo de él.

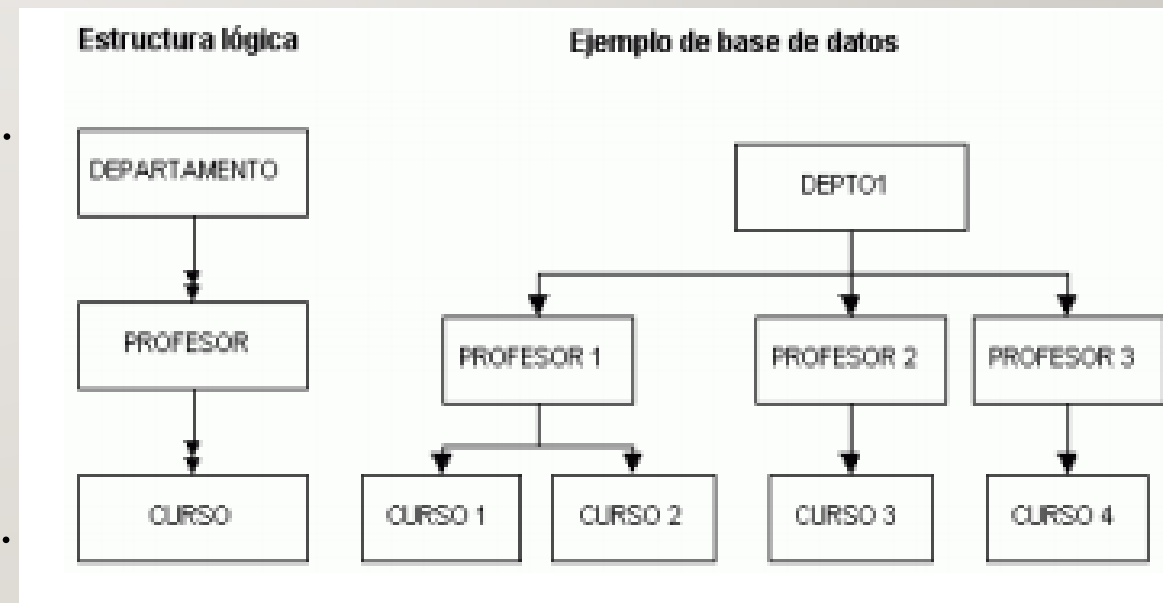
Las bases de datos jerárquicas son especialmente útiles en el caso de aplicaciones que manejan un gran volumen de información y datos muy compartidos permitiendo crear estructuras estables y de gran rendimiento.

Una de las principales limitaciones de este modelo es su incapacidad de representar eficientemente la redundancia de datos.



La representación gráfica de este modelo se realiza mediante la creación de un árbol invertido, los diferentes niveles quedan unidos mediante relaciones. En este modelo solo se pueden representar relaciones 1:M, por lo que presenta varios inconvenientes:

- No se admiten relaciones N:M
- Un segmento hijo no puede tener más de un padre.
- No se permiten más de una relación entre dos segmentos.
- Para acceder a cualquier segmento es necesario comenzar por el segmento raíz.
- El árbol se debe de recorrer en el orden designado.

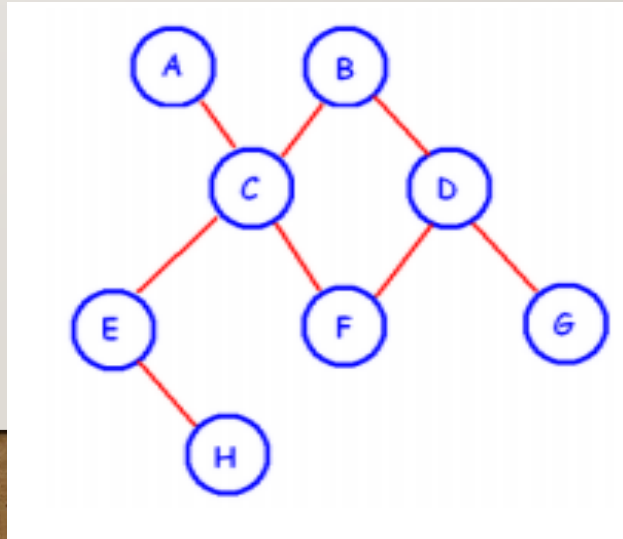


MODELO DE RED

En esta estructura cualquier componente puede relacionarse con cualquier otro.

A diferencia del modelo jerárquico, en este modelo, un hijo puede tener varios padres.

Agregado de datos, que define un conjunto de datos con nombre. Este modelo de datos permite representar relaciones N:M Aquí se representa los datos mediante colecciones de registros y sus relaciones se representan por medio de ligas o enlaces, los cuales pueden verse como punteros.



BASES DE DATOS ORIENTADAS A OBJETOS

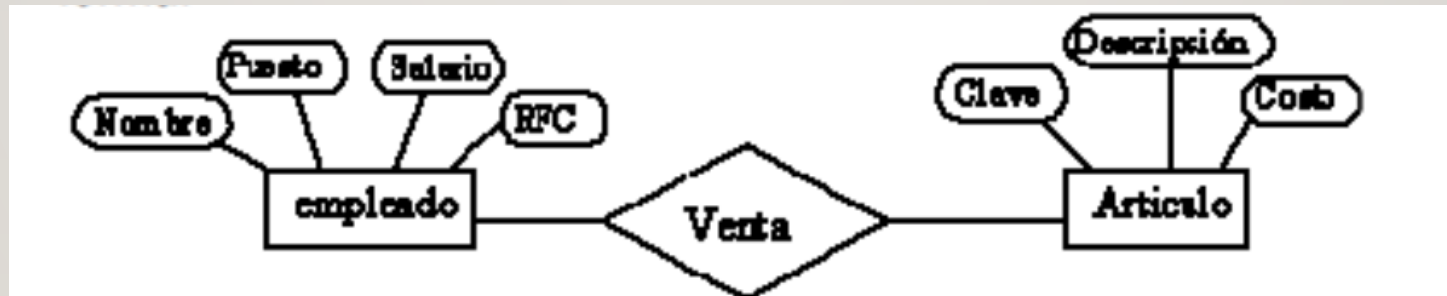
- Capacidad de manipular objetos, se basan en la existencia de objetos persistentes que se almacenan para su procesamiento.
- Estas bases de datos almacenan colecciones de objetos mismos que tienen comportamiento.

BASES DE DATOS DISTRIBUIDAS

- Consisten en multiplicar el número de servidores que controlan una base de datos (se les llama nodos), intercambiando y actualizando información a través de una red.

MODELO ENTIDAD RELACIÓN

Este modelo representa a la realidad a través de entidades, que son objetos que existen y que se distinguen de otros por sus características, por ejemplo: un alumno se distingue de otro por sus características particulares como lo es el nombre, o el numero de boleta asignado al entrar a una escuela, así mismo, un empleado, una materia, etc.



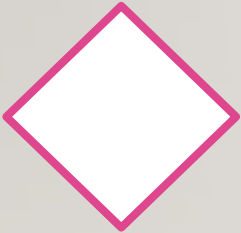
COMPONENTES



Entidad



Atributo



Relación



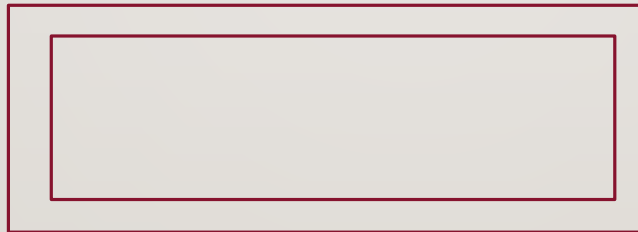
Liga

ENTIDAD

Se trata de cualquier objeto u elemento (real o abstracto) acerca del cual se pueda almacenar información en la base de datos. Ejemplos de entidades son, la factura número 32456, el coche matrícula 3452BCW.

TIPOS DE ENTIDADES

- Regulares. Son las entidades normales que tienen existencia por sí mismas sin depender de otras. Su representación gráfica es la indicada arriba □
- Débiles. Su existencia depende de otras. Por ejemplo la entidad Actividades sólo podrá tener existencia si existe la entidad puesto.



RELACIONES

- Representan asociaciones entre entidades. Es el elemento del modelo que permite relacionar en sí los datos del modelo. Por ejemplo, en el caso de que tengamos una entidad personas y otra entidad trabajos. Ambas se realizan ya que las personas trabajan y los trabajos son realizados por personas.

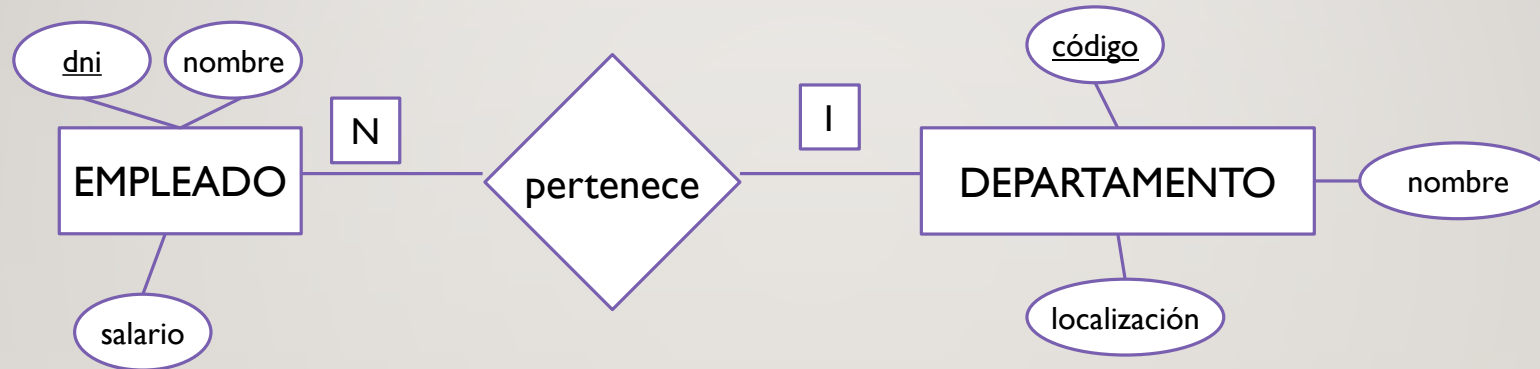
REPRESENTACIÓN GRÁFICA

- La representación gráfica de las entidades se realiza con un rombo al que se le unen líneas que se dirigen a las entidades, las relaciones tienen nombre (se suele usar un verbo). En el ejemplo anterior podría usarse como nombre de relación, trabajar:

CARDINALIDAD

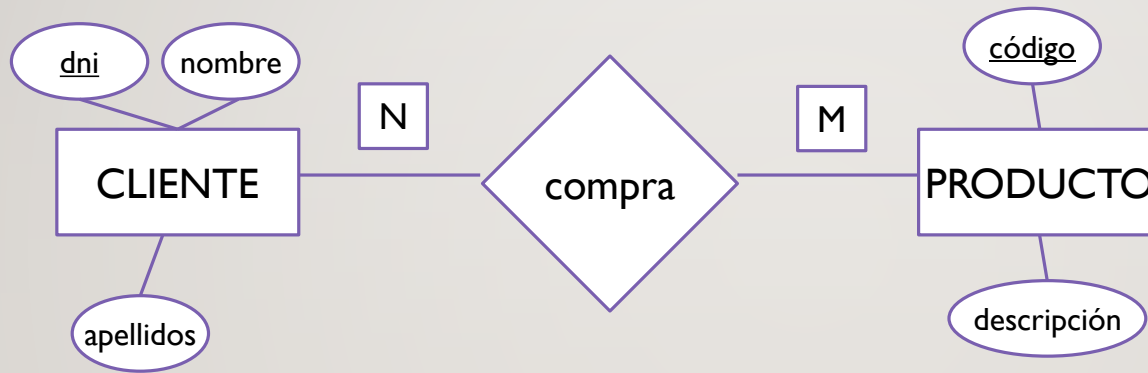
- Indica el número de relaciones en las que una entidad puede aparecer. Se anota en términos de:
- Cardinalidad mínima. Indica el número mínimo de asociaciones en las que aparecerá cada ejemplar de la entidad (el valor que se anota es de cero o uno)
- Cardinalidad máxima. Indica el número máximo de relaciones en las que puede aparecer cada ejemplar de la entidad (puede ser uno o muchos)

EJEMPLO: RELACIONES N:I



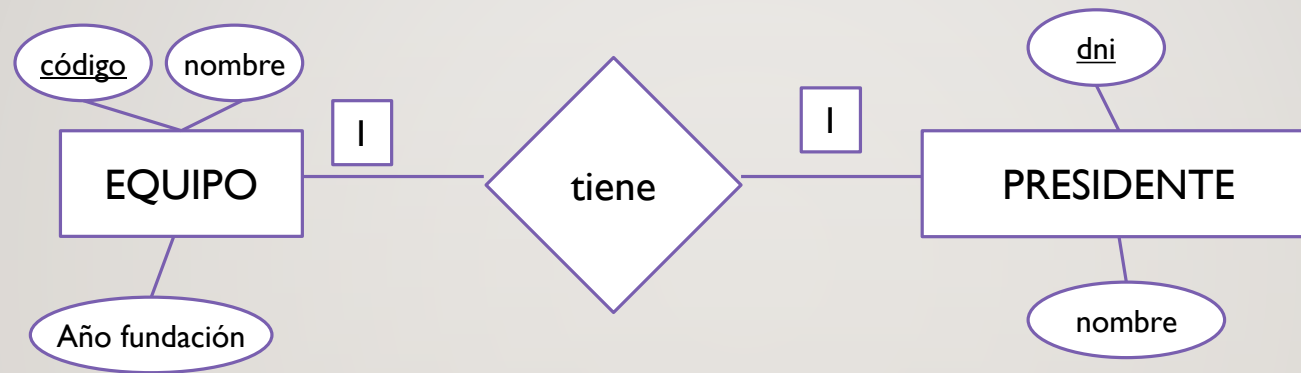
- EMPLEADO (**dni**, nombre, salario, código_departamento)
- DEPARTAMENTO (**código**, nombre, localización)

EJEMPLO: RELACIONES DE N:M (MUCHOS A MUCHOS)



- CLIENTE (**dni**, nombre, apellidos)
- PRODUCTO (**código**, descripción)
- COMPRAS (**dni_cliente**, **código _ producto**, fecha_compra)

EJEMPLO 3: RELACIONES 1:1



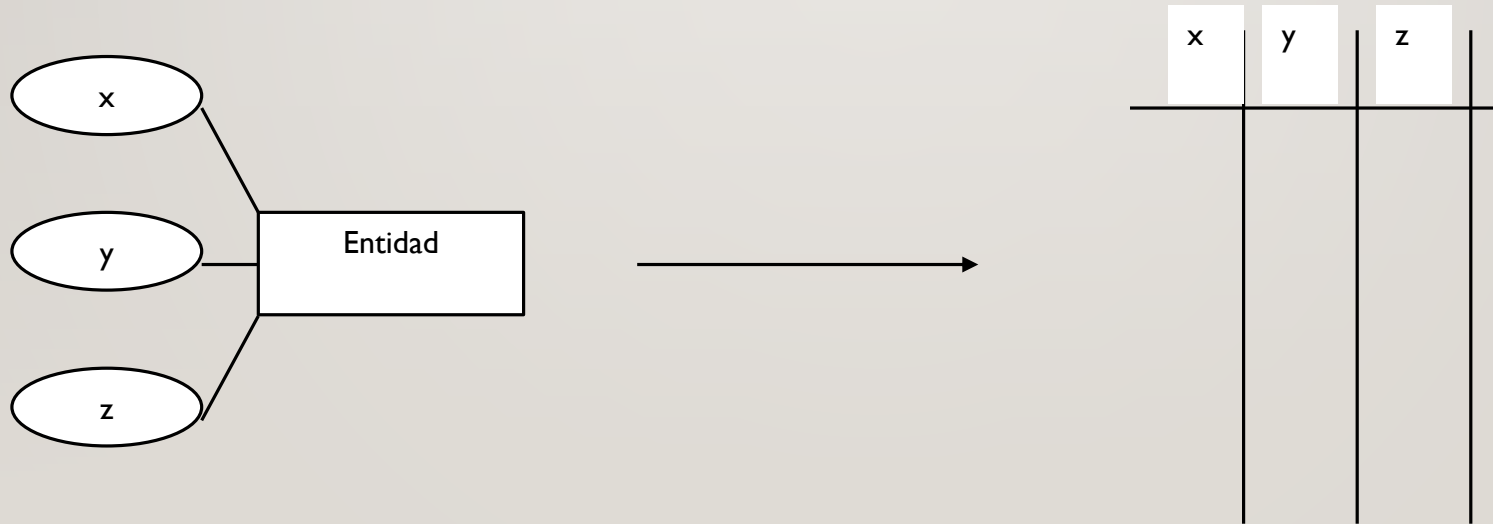
- EQUIPO (**código**, nombre, año _ fundación)
- PRESIDENTE (**dni**, nombre, código _ equipo)

MODELO RELACIONAL

- Descripción de datos Entidades y relaciones se representan en forma de tablas:
- Las tablas reciben el nombre de relación.
- Las filas (tuplas) contienen datos sobre cada entidad.
- Las columnas corresponden a los atributos de las entidades.

-
- Relación: Conjunto de atributos, cada uno de los cuales pertenece a un dominio, y que posee un nombre que identifica la relación.

En este modelo la base de datos es vista por el usuario como una relación de tablas. Cada fila de la tabla es un registro o tupla y los atributos con columnas o campos.



CONCEPTOS DE BASES DE DATOS RELACIONALES:

Relación: Película (título, año, duración)

Atributos →	Título	Año	Duración
	La guerra de las galaxias	1977	123
Tuplas ↗ ↘	El señor de los anillos I	2001	178

↓
Dominio=textos

↘ ↙
Dominio=enteros

Cardinalidad= 2

Grado de la relación= 3

Relación= Conjunto ordenado de n ocurrencias

Atributos= Campos de una tabla, propiedades de las entidades

Dominio= Conjunto donde los atributos toman valores

Tupla= Fila de una tabla

Grado de una relación= Numero de atributos o columnas

Cardinalidad= Numero de filas o tuplas de una relación

Clave primaria= Es un conjunto de atributos que identifica a cada tupla de una relación y además no hay un subconjunto de ellos que cumplan esa propiedad.

Clave foránea= Es un conjunto de atributos de una tabla que son clave primaria en otra tabla

EJEMPLO

Proveedores

Código	Nombre	Ciudad
S1	Juan Gómez	Alicante
S2	Raquel Jiménez	Alicante
S3	Pablo Fernández	Madrid
S4	Carmen López	Granada
S5	Cristina García	Madrid

Piezas

Código	Nombre	Dimensiones	Peso
P1	Tuerca	12 mm	1.2 gr
P2	Tornillo	10 mm	1.7 gr
P3	Tornillo	20 mm	2.7 gr
P4	Clavo	25 mm	1.4 gr
P5	Arandela	15 mm	3.0 gr
P6	Tuerca	10 mm	1.0 gr

CARDINALIDAD

- Calcula la cardinalidad de las siguientes relaciones binarias:
- Hombre está casado con Mujer, en una sociedad monogámica
- Hombre está casado con Mujer, en una sociedad poligámica
- Pescador pesca Pez
- Piezas forman Casa

DOMINIO

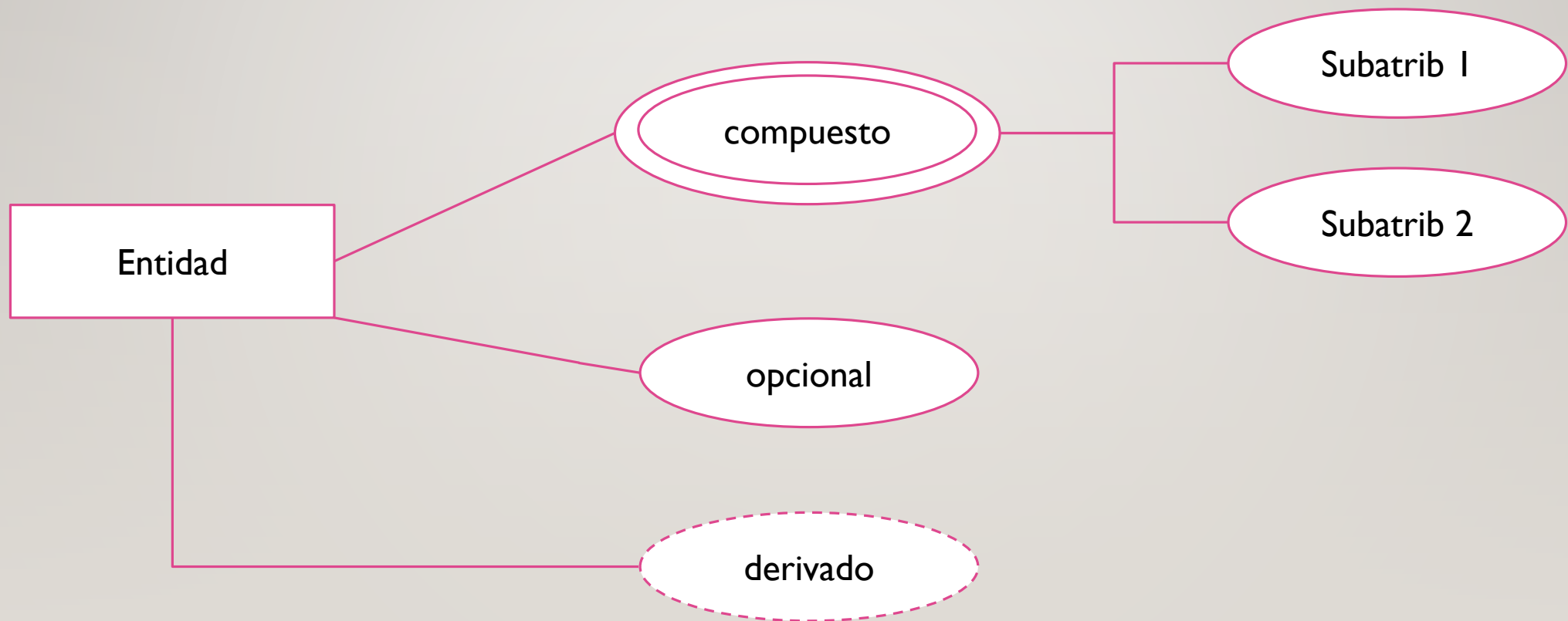
El dominio representa la naturaleza del dato.

Ejemplo:

Atributo	Dominio
IDEmpleado	Cadena de caracteres (5)
Nombre	Cadena de caracteres (50)
Dirección	Cadena de caracteres (100)
Sueldo	Números reales
FechaNacimiento	Fecha

ATRIBUTOS

- Atributos obligatorios: Un atributo debe tomar un valor obligatoriamente.
- Atributos opcionales: Un atributo puede no tomar un valor por que sea desconocido, es decir puede ser nulo.
- Atributos compuestos: Es aquel que se puede descomponer en atributos mas sencillos, por ejemplo, el atributo horaSalida se puede descomponer en dos (hora y minutos).
- Atributo derivado: Son aquellos cuyo valor se puede calcular a través de otros atributos. Por ejemplo, el atributo Edad, se puede calcular a partir de la fecha de nacimiento de una persona.



EJEMPLO

Una empresa vende productos a varios clientes.

- Se necesita conocer los datos personales de los clientes (nombre, apellidos, id, dirección y fecha de nacimiento).
- Cada producto tiene un nombre y un código, así como un precio unitario.
- Un cliente puede comprar varios productos a la empresa, y un mismo producto puede ser comprado por varios clientes.
- Los productos son suministrados por diferentes proveedores.
- Se debe tener en cuenta que un producto sólo puede ser suministrado por un proveedor, y que un proveedor puede suministrar diferentes productos.
- De cada proveedor se desea conocer el ID, nombre y dirección”.

REALIZAR EL DIAGRAMA ENTIDAD RELACIÓN Y MODELO RELACIONAL DE LOS SIGUIENTES EJERCICIOS:

I. Una sala de belleza desea llevar el control de sus empleadas y sus clientes, así como de los servicios que se prestan y la venta de cosméticos. Se desea almacenar la siguiente información:

- Empleadas: id, Nombre, especialidad (corte, color, permanente, maquillaje).
- Clientes: Datos personales (id, nombre, teléfono y dirección)
- Servicios: Que empleada atendió a que qué cliente, y que tipo de servicio le prestó en que fecha y hora.
- Cosmético: Código, nombre, cantidad y precio.
- Ventas de cosméticos: una empleada vende un cosmético a un cliente.

EJERCICIOS

2. Se desea diseñar una base de datos para almacenar y gestionar la información empleada por una empresa dedicada a la venta de automóviles, teniendo en cuenta los siguientes aspectos:

- La empresa dispone de una serie de coches para su venta. Se necesita conocer la matrícula, marca y modelo, el color y el precio de venta de cada coche.
- Los datos que interesa conocer de cada cliente son el id, nombre, dirección, ciudad y número de teléfono: además, los clientes se diferencian por un código interno de la empresa que se incrementa automáticamente cuando un cliente se da de alta en ella.
- Un cliente puede comprar tantos coches como desee a la empresa.
- Un coche determinado solo puede ser comprado por un único cliente.
- El concesionario se encarga de llevar a cabo las revisiones que se realizan a cada coche.
- Cada revisión tiene asociado un código que se incrementa automáticamente por cada revisión que se haga.
- De cada revisión se desea saber si se ha hecho cambio de filtro, si se ha hecho cambio de aceite, si se ha hecho cambio de frenos u otros.
- Los coches pueden pasar varias revisiones en el concesionario.



3. Se desea automatizar la gestión de una empresa de transportes que reparte paquetes por toda la república.

Los encargados de llevar los paquetes son los mensajeros, de los que se quiere guardar el id, nombre, teléfono, dirección, salario y población en la que vive.

De los paquetes transportados interesa conocer el código de paquete, descripción, destinatario y dirección del destinatario.

Un mensajero distribuye muchos paquetes, y un paquete sólo puede ser distribuido por un mensajero.

De los estados a los que llegan los paquetes interesa guardar el código y el nombre.

Un paquete sólo puede llegar a un estado. Sin embargo, a un estado pueden llegar varios paquetes.

De los autos que llevan los mensajeros, interesa conocer la matrícula, modelo, tipo y potencia.

Un mensajero puede conducir solo un auto.



4. En la biblioteca del centro se manejan fichas de autores y libros.

- En la ficha de cada autor se tiene el código de autor y el nombre.
- De cada libro se guarda el código, título, ISBN, editorial y número de página.
- Un autor puede escribir varios libros, y un libro puede ser escrito por varios autores.
- Un libro tiene muchos ejemplares y un ejemplar pertenece sólo a un libro. Cada ejemplar tiene un código y una localización.
- Los usuarios de la biblioteca del centro también disponen de ficha en la biblioteca y sacan ejemplares de ella. De cada usuario se guarda el código, nombre, dirección y teléfono.
- Los ejemplares son prestados a los usuarios. Un usuario puede tomar prestados varios ejemplares, y un ejemplar puede ser prestado a varios usuarios.
- De cada préstamo interesa guardar el usuario, código de ejemplar, la fecha de préstamo, la fecha de devolución.

