



BASE DE DATOS

Teoría

ÍNDICE

	Página
Presentación	5
Red de contenidos	7
Unidad de Aprendizaje 1	
SISTEMA DE BASE DE DATOS	
1.1 Tema 1 : Introducción al curso	10
1.1.1 : Importancia del curso dentro de la carrera	10
1.2 Tema 2 : Conceptos básicos	10
1.2.1 : Surgimiento histórico de las bases de datos	10
1.2.2 : Definición de la base de datos	12
1.2.3 : Objetivos de la base de datos	12
1.2.4 : Herramientas de modelado (DER): yED, Entorno de yED.	14
1.2.5 : Obtención de un DER básico	15
1.3 Tema 3 : Representación de la información	17
1.3.1 : Niveles de abstracción referidos a la información	17
1.3.2 : Reglas de negocio	20
1.3.3 : Relaciones de correspondencia	20
1.3.4 : Ejercicios de aplicación	23
1.3.5 : Autoaprendizaje	24
1.4 Tema 4 : Arquitectura de un sistema de base de datos	27
1.4.1 : Arquitectura de un sistema de base de datos	27
1.4.2 : Organizaciones de archivos y el nivel interno de la arquitectura	30
Unidad de Aprendizaje 2	
MODELO CONCEPTUAL	
2.1 Tema 5 : Modelo conceptual	37
2.1.1 : Modelo conceptual de datos	37
2.2 Tema 6 : Diagrama Entidad-Relación	40
2.2.1 : Entidades: Generalización / Especialización	43
2.2.2 : Entidad fuerte/débil	47
2.2.3 : Ejercicios de aplicación	48
2.2.4 : Relaciones recursivas	49
2.2.5 : Agregaciones	50
2.2.6 : Ejercicios integradores	52
Unidad de Aprendizaje 3	
MODELO RELACIONAL Y NORMALIZACIÓN	
3.1. Tema 7 : Normalización parte I	68
3.1.1 : Modelo relacional	68
3.1.2 : Primera Forma Normal	75
3.1.3 : Segunda Forma Normal	82
3.1.4 : Tercera Forma Normal	83
3.1.5 : Integración: Ejercicios propuestos	87
3.1.6 : Ejercicios de normalización básicos	89
3.2. Tema 8 : Normalización parte II	91
3.2.1 : Sin grupos repetitivos	91
3.2.2 : Con un grupo repetitivo	93

3.2.3 : Ejercicios	93
3.2.4 : Múltiples grupos repetitivos	96
3.2.5 : Ejercicios	96
3.2.6 Diccionario de datos	99
3.3. Tema 9 : Normalización parte III	99
3.3.1 : Normalización Detalle del Detalle / Ítems	100
3.3.2 : Ejercicios parte I	100
3.3.3 : Ejercicios propuestos: Detalle del Detalle / Ítems Detalle	107
3.3.4 : Ejercicios propuestos (adicionales)	110

Unidad de Aprendizaje 4**DIAGRAMA ENTIDAD RELACIÓN**

4.1. Tema 10 : Modelo lógico (DER a MER)	116
4.1.1 : Obtención del modelo lógico-global de los datos a partir del DER	116
4.1.2 : Modelo lógico. Ejercicios	119
4.2. Tema 11 : Modelo físico (MER a DER)	125
4.2.1 : Metodología de sistema de base de datos	126
4.2.2 : Obtención del modelo lógico-global de los datos a partir del MER	134

PRESENTACIÓN

El presente manual está diseñado para que sea utilizado como herramienta de trabajo en el curso de Bases de Datos. Por esto, se encuentra estructurado de acuerdo con la organización del mismo.

Cada sesión contempla, entre otros aspectos, los logros que se persiguen y los contenidos que se abordan en la misma, lo cual permite que el alumno pueda tener una idea clara de los aspectos que se van a tratar y del alcance de éstos en el curso.

En el desarrollo de los contenidos, las explicaciones están ilustradas con ejemplos y figuras que facilitan la comprensión. Algunas sesiones están completamente dedicadas a la ejercitación de los contenidos impartidos, de modo que los alumnos puedan alcanzar habilidades en la aplicación de las técnicas estudiadas.

Al final de cada sesión, se añade una Autoevaluación, en la cual se proponen ejercicios para que los estudiantes los realicen independientemente, fuera de clases, de manera que puedan aumentar sus habilidades y así comprobar sus conocimientos. Para recordar, brinda, de modo resumido, las conclusiones más importantes acerca de los contenidos tratados.

RED DE CONTENIDOS

Base de Datos

Unidad 1: Sistema de base de datos

- Introducción al curso
- Conceptos básicos
- Representación de la información
- Arquitectura de un sistema de base de datos

Unidad 2: Modelo Conceptual

- Modelo conceptual de datos
- Diagrama Entidad - Relación

Unidad 3: Modelo relacional y Normalización

- Normalización parte I
- Normalización parte II
- Normalización parte III

Unidad 4: Diagrama Entidad Relación

- Modelo lógico (DER a MER)
- Modelo físico (MER a DER)



INTRODUCCIÓN AL CURSO

LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la primera unidad, el estudiante describe los componentes de una base de datos a partir de casos propuestos por el profesor, tomados de situaciones cotidianas y de ejemplos producidos individualmente. Asimismo, determina los elementos de una base de datos y sus interrelaciones.

Temario

1.1 Tema 1 : Introducción al curso

1.1.1 : Importancia del curso dentro de la carrera

1.2 Tema 2 : Conceptos básicos

1.2.1 : Surgimiento histórico de las bases de datos

1.2.2 : Definición de la base de datos

1.2.3 : Objetivos de la base de datos

1.2.4 : Herramientas de modelado (DER): yED, Entorno de yED.

1.2.5 : Obtención de un **DER** básico

1.3 Tema 3 : Representación de la información

1.3.1 : Niveles de abstracción referidos a la información

1.3.2 : Reglas de negocio

1.3.3 : Relaciones de correspondencia

1.3.4 : Ejercicios de aplicación

1.3.4 : Autoaprendizaje

1.4 Tema 4 :

1.4.1 :

1.4.2 : Organizaciones de archivos y el nivel interno de la arquitectura

1.4.3 : Autoaprendizaje

ACTIVIDADES PROPUESTAS

1. Explicaciones por parte del facilitador combinadas con ejemplos.
2. Formulaciones de preguntas de comprobación dirigidas a los alumnos sobre los temas tratados.

1.1. Tema 1: Introducción al curso

1.1.1. Importancia del curso dentro de la carrera

En un mundo globalizado, las Bases de Datos tienen una gran relevancia a nivel personal, pero más a nivel empresarial; ya que se consideran una de las mayores aportaciones que ha dado la informática a las empresas. En la actualidad, cualquier organización, por más pequeña que sea, debe contar con una Base de Datos y no basta tenerla; sino debemos saber cómo gestionarlas. Es por ello la importancia de incluir dentro de la carrera. El objetivo del curso es que los estudiantes logren sólidos conocimientos sobre los temas a tratar para involucrarse con seguridad en administrar una base de datos del mundo real.

1.2. Tema 2 : Conceptos Básicos

1.2.1. Surgimiento histórico de las bases de datos

Al estudiar el desarrollo del procesamiento automatizado de datos, en lo que se refiere al aseguramiento técnico, se habla de diferentes generaciones.

Desde el punto de vista del aseguramiento matemático y, en particular, del aseguramiento de programas, algunos autores reconocen **tres generaciones**:

- Solución de tareas aisladas
- Integración de tareas aisladas en sistemas particulares
- Integración de sistemas particulares en sistemas automatizados de dirección

Este proceso de integración ocurre paralelamente, aunque no simultáneamente, en dos esferas:

a) Integración de los programas

Los lenguajes de programación integran a las bases de datos con un propósito específico, el cual se manifiesta en sus aplicaciones **Cliente-Servidor**. Es así que en la actualidad los lenguajes de programación presentan un entorno sofisticado de desarrollo que permiten trabajar de manera integrada con las bases no necesariamente de un determinado proveedor de datos.

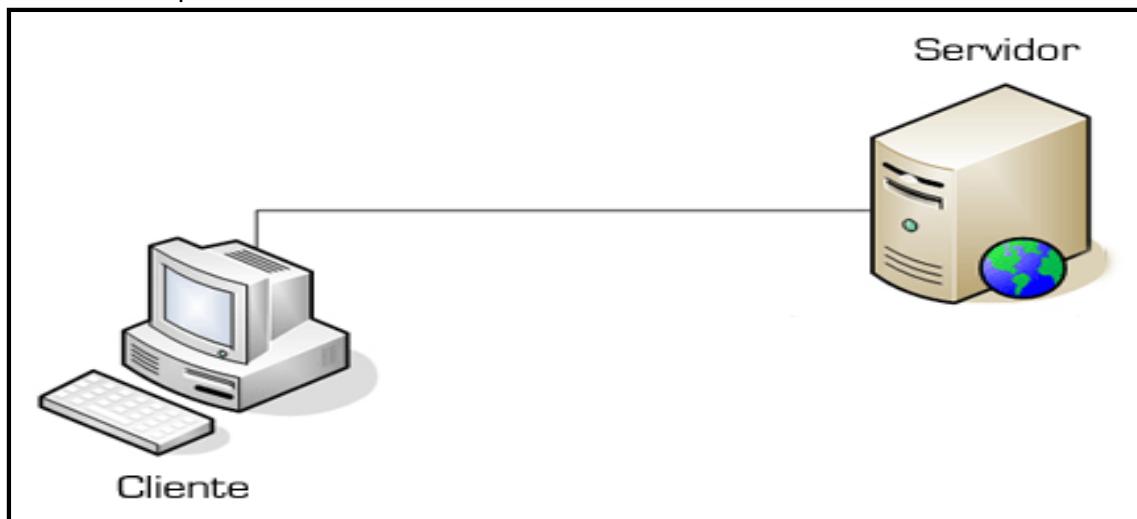


Figura 1: Cliente-Servidor

Fuente.- Tomado de <http://www.coders.me/wp-content/uploads/2008/03/cliente-servidor.gif>

b) Integración de los datos

Se han producido tres categorías de técnicas para su manipulación:

- **Sistemas orientados a los dispositivos**

Programas y archivos que son diseñados y empleados de acuerdo con las características físicas de la unidad central y los periféricos. Cada programa está altamente interconectado con sus archivos, por lo que la integración de datos de diferentes sistemas es imposible prácticamente.

- **Sistemas orientados a los archivos**

La lógica de los programas depende de las técnicas de organización de los archivos (secuencial, directo, etc.). Cada usuario organiza su archivo de acuerdo con sus necesidades y las relaciones entre los elementos se establecen a través de los programas de aplicación.

Esta forma de trabajo implica redundancia de datos, que trae aparejada mayor gasto de memoria y complica las operaciones de actualización (modificar un dato donde quiera que aparezca). Esto aumenta el tiempo de tratamiento y atenta contra la integridad de la información. Cuando se habla de integridad, se está haciendo referencia a que, en todo momento, los datos almacenados estén correctos en correspondencia con la realidad.

Además, en la vida real, se establecen relaciones entre los objetos que son muy difíciles de representar u obtener a partir de sistemas tradicionales de archivos. Por ejemplo, si se tiene información sobre trabajadores y estudiantes de una facultad, las aplicaciones requeridas van a definir la manera de organizar y estructurar los archivos. Si se desean obtener datos como promedio de las calificaciones de cada alumno, listado de estudiantes por grupo, categoría científica y docente de cada grupo, y salario de cada uno, resulta adecuado establecer dos archivos: uno de profesores y otro de estudiantes.

¿Qué ocurre si se quieren establecer vínculos entre los profesores y los estudiantes?

Por ejemplo, si se desea obtener lo siguiente:

- Los estudiantes de un profesor
- Los profesores de un estudiante

De esta manera, se estructuraría un archivo de profesores y estudiantes que resolvería algunas demandas, pero sería ineficiente para otras.

Entonces.

¿Es posible representar de manera eficiente, utilizando los medios de cómputo, los casos o procesos de la realidad objetiva (procesos de negocio), aunque sea, por supuesto, de forma esquemática, pero en la que se establezcan determinados vínculos entre los elementos u objetos que forman parte de esos procesos o casos?

Se observa que es posible hacerlo a través de la utilización de bases de datos (**BD**) y de los sistemas de gestión de bases de datos (**SGBD**) que dirigen su manipulación.

- **Sistemas orientados a base de datos**

Un Sistema de Gestión de Bases de Datos consiste en una colección de datos interrelacionados y un conjunto de programas para acceder a los mismos.

1.2.2. Definición de la base de datos

Veamos algunas definiciones:

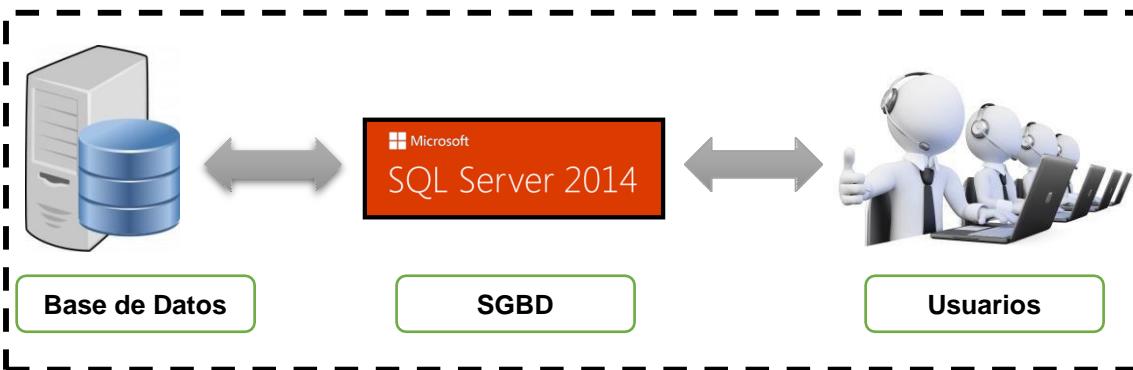
“Se define una base de datos como una serie de datos organizados y relacionados entre sí, los cuales son recolectados y explotados por los sistemas de información de una empresa o negocio en particular”.

“Se le llama base de datos a los bancos de información que contienen datos relativos a diversas temáticas y categorizados de distinta manera, pero que comparten entre sí algún tipo de vínculo o relación que busca ordenarlos y clasificarlos en conjunto”.

“Una base de datos es un “almacén” que nos permite guardar grandes cantidades de información de forma organizada con el objeto de encontrar y utilizar fácilmente dicha información”.

“Una base de datos correctamente diseñada permite obtener acceso a información exacta y actualizada. Puesto que un diseño correcto es esencial para lograr los objetivos fijados para la base de datos, parece lógico emplear el tiempo que sea necesario en aprender los principios de un buen diseño ya que, en ese caso, es mucho más probable que la base de datos termine adaptándose a sus necesidades y pueda modificarse fácilmente”. Microsoft.

El software que permite la utilización y/o la actualización de los datos almacenados en una (o varias) base(s) de datos por uno o varios usuarios desde diferentes puntos de vista y a la vez se denomina sistema de gestión de bases de datos (**SGBD**). Es importante diferenciar los términos base de datos y **SGBD**. El objetivo fundamental de un **SGBD** consiste en suministrar al usuario las herramientas que le permitan manipular, en términos abstractos, los datos, de tal forma que no le sea necesario conocer el modo de almacenamiento de los datos en la computadora ni el método de acceso empleado. Los programas de aplicación operan sobre los datos almacenados en la base, utilizando las facilidades que brindan los **SGBD**, los que, en la mayoría de los casos, poseen lenguajes especiales de manipulación de la información que facilitan el trabajo de los usuarios.



1.2.3. Objetivos de la base de datos

Existen muchas formas de organizar las bases de datos, pero hay un conjunto de objetivos generales que deben cumplir todos los **SGBD**, de modo que faciliten el proceso de diseño de aplicaciones y que los tratamientos sean más eficientes y rápidos, dando la mayor flexibilidad posible a los usuarios.

Los objetivos fundamentales de los **SGBD** son los siguientes:

1. Independencia de los datos y los programas de aplicación

Se ha observado que, con archivos tradicionales, la lógica de la aplicación contempla la organización de los archivos y el método de acceso. Por ejemplo, si por razones de eficiencia se utiliza un archivo secuencial indexado, el programa de aplicación debe considerar la existencia de los índices y la secuencia del archivo. Entonces, es imposible modificar la estructura de almacenamiento o la estrategia de acceso sin afectar el programa de aplicación (naturalmente, lo que se afecta en el programa son las partes de éste que tratan los archivos, lo que es ajeno al problema real que el programa de aplicación necesita resolver). En un **SGBD**, sería indeseable la existencia de aplicaciones y datos dependientes entre sí, por dos razones fundamentales:

- a) Diferentes aplicaciones necesitarán diferentes aspectos de los mismos datos (por ejemplo, puede requerirse la representación decimal o binaria).
- b) Se debe modificar la estructura de almacenamiento o el método de acceso, según los cambios en el caso o proceso de la realidad sin necesidad de modificar los programas de aplicación (también para buscar mayor eficiencia).

La independencia de los datos se define como la inmunidad de las aplicaciones a los cambios en la estructura de almacenamiento y en la estrategia de acceso. Todo esto constituye el objetivo fundamental de los **SGBD**.

2. Minimización de la redundancia

Se ha comprobado cómo, con los archivos tradicionales, se produce redundancia de la información. Uno de los objetivos de los **SGBD** es minimizar la redundancia de los datos. Se dice disminuirla, no eliminarla, pues, aunque se definen las bases de datos como no redundantes, en realidad sí existe, pero en un grado no significativo que servirá para disminuir el tiempo de acceso a los datos o para simplificar el método de direccionamiento. Lo que se trata de lograr es la eliminación de la redundancia superflua.

3. Integración y sincronización de las bases de datos

La integración consiste en garantizar una respuesta a los requerimientos de diferentes aspectos de los mismos datos por diferentes usuarios, de forma que, aunque el sistema almacene la información con cierta estructura y cierto tipo de representación, debe garantizar entregar al programa de aplicación los datos que solicita y en la forma en que lo solicita.

Está vinculada a la sincronización, que consiste en la necesidad de garantizar el acceso múltiple y simultáneo a la base de datos, de modo que puedan ser compartidos por diferentes usuarios a la vez. Están relacionadas, ya que lo usual es que diferentes usuarios trabajen con diferentes enfoques y requieran los mismos datos, pero desde diferentes puntos de vista.

4. Integridad de los datos

Consiste en garantizar la no contradicción entre los datos almacenados, de modo que, en cualquier momento del tiempo, sean correctos, es decir, que no se detecte inconsistencia entre los mismos. Está relacionada con la minimización de la redundancia, ya que es más fácil garantizar la integridad si se elimina ésta.

5. Seguridad y recuperación

- ✓ **Seguridad** (también llamada protección): garantiza el acceso autorizado a los datos, la forma de interrumpir cualquier intento de acceso no autorizado, ya sea por error del usuario o por mala intención.
- ✓ **Recuperación:** permite que el sistema de bases de datos disponga de métodos que garanticen la restauración de las bases de datos al producirse alguna falla técnica, interrupción de la energía eléctrica, etc.

6. Facilidad de manipulación de la información

Los usuarios de una base de datos pueden acceder a ella con solicitudes para resolver muchos problemas diferentes. El **SGBD** debe contar con la capacidad de una búsqueda rápida por diferentes criterios, debe permitir que los usuarios planteen sus demandas de una forma simple, aislándolo de las complejidades del tratamiento de los archivos y del direccionamiento de los datos. Los **SGBD** actuales brindan lenguajes de alto nivel, con diferentes grados de facilidad para el usuario no programador, que garantizan este objetivo, los llamados sublenguajes de datos.

7. Control centralizado

Uno de los objetivos más importantes de los **SGBD** es garantizar el control centralizado de la información. Permite comprobar, de manera sistemática y única, los datos que se almacenan en la base de datos, así como el acceso a ella.

Lo anterior implica que debe existir una persona o un conjunto, que tenga la responsabilidad de los datos operacionales: el administrador de la base de datos puede considerarse parte integrante del **SGBD**.

1.2.4. Herramientas de Modelado (DER): yED, Entorno de yED.

Existen una variedad de herramientas que nos facilita de manera sencilla y eficaz elaborar el modelo de Diagrama entidad Relación de una base de datos. A continuación, podemos mencionar como, por ejemplo:

- CASE Studio 2
 - link: <http://www.casestudio.com/enu/index.aspx>
- DIA
 - link: <https://dia.uptodown.com/windows/all>
- Yed Graph Editor
 - link: <https://www.yworks.com/products/yed>
- Etc.

En el curso utilizaremos la herramienta **yED Graph Editor**

¿Qué es yED?

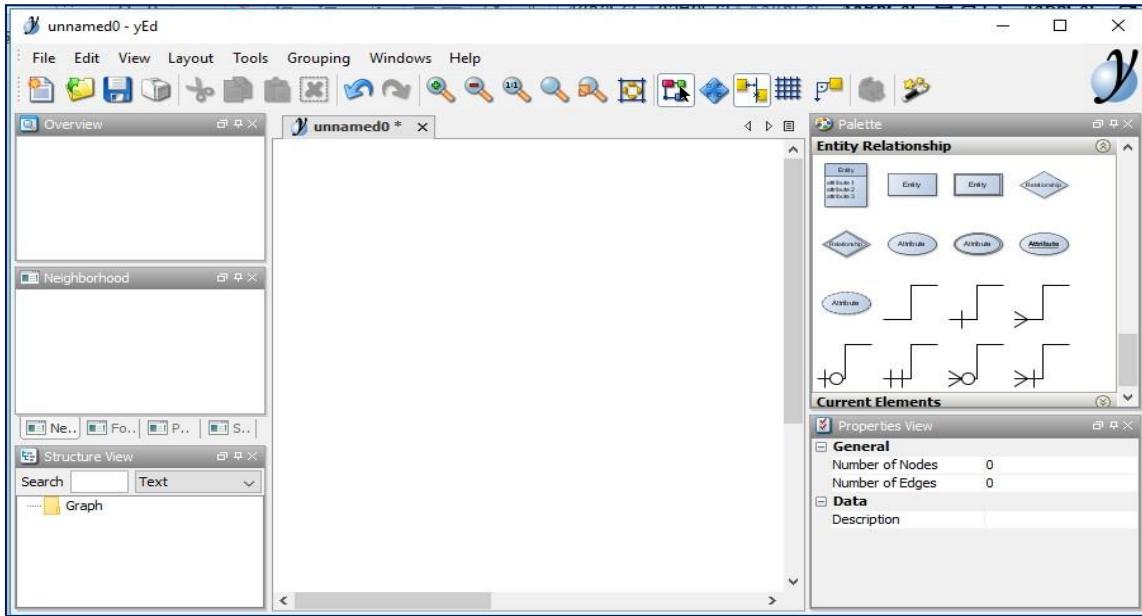
Es una poderosa aplicación de escritorio que se puede utilizar para generar de forma rápida y eficaz diagramas de alta calidad.

yED está libremente disponible y funciona en todas las plataformas: Windows, UNIX / Linux, y Mac OS X.



yEd 3.16.2.1

Entorno de yED



1.2.5. Obtención de un DER básico

CASO 01: Registro de Automóvil

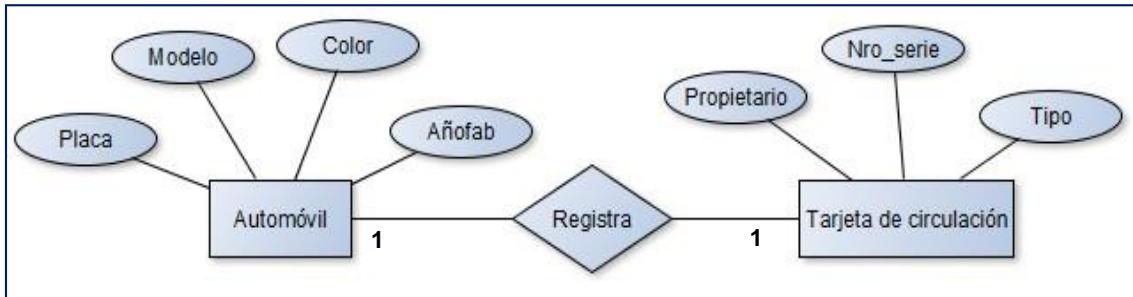
Se requiere obtener la tarjeta de circulación de un automóvil tomando en cuenta los siguientes datos:

- Automóvil: Modelo, placa, color y año de fabricación.
- Tarjeta de circulación: Propietario, nro_serie y tipo.

Realizar el diagrama ENTIDAD/RELACIÓN para este sistema de información indicando las entidades, atributos, relaciones y sus características

Nota: utilizaremos las simbologías básicas para la elaboración.

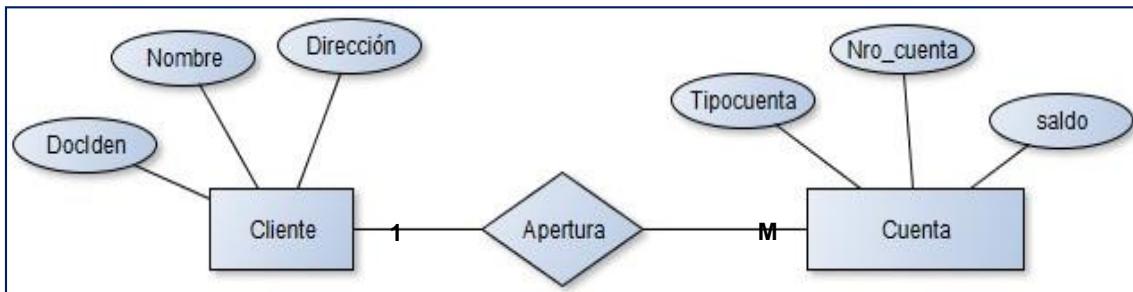
SIMBOLOGÍA UTILIZADA EN EL DIAGRAMA ENTIDAD / RELACIÓN		
SÍMBOLO	DESCRIPCIÓN	EJEMPLO
Entidad 	Rectángulos: representan conjuntos de Entidades.	CLIENTE
Atributo 	Elipses: representan atributos	Nombre
Conexión 	Líneas: conectan los atributos a los conjuntos de entidades, y los conjuntos de relaciones	
Relación 	Rombos: representan relaciones.	

Solución:**CASO 02: Cuentas**

El banco “**Préstame ya**” desea controlar la información de sus clientes con sus cuentas correspondientes. Para ello toma en cuenta los siguientes datos:

- Cliente: Documento de Identidad, nombre y dirección.
- Cuenta: Tipocuenta, nro_cuenta y saldo.

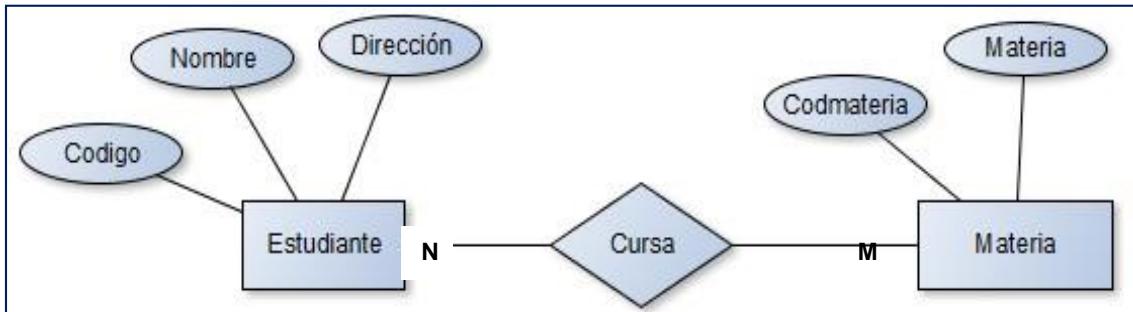
Además, un cliente puede tener muchas cuentas, pero que una cuenta puede llegar a pertenecer a un solo cliente (Se indica que puede, ya que existen cuentas registradas a favor de más de una persona “En este caso”).

Solución:**CASO 03: Centro Educativo**

El centro educativo “**ANE**” desea controlar la información de sus estudiantes; así como la materia que está llevando. Para ello toma en cuenta los siguientes datos:

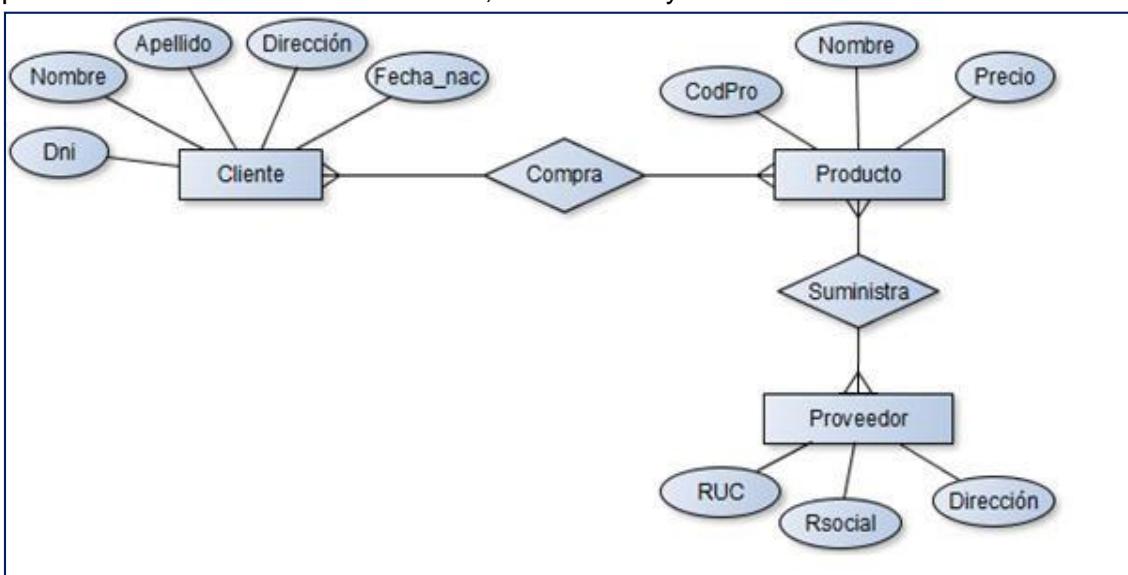
- Estudiante: Código, nombre y dirección.
- Materia: Código de materia, materia.

Además, un estudiante puede cursar muchas materias, y una materia puede ser cursada por muchos estudiantes.

Solución:

CASO 04: Compra

Una empresa vende productos a varios clientes. Se necesita conocer los datos personales de los clientes (nombre, apellidos, DNI, 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 puede ser suministrado por varios proveedores, y los proveedores pueden suministrar diferentes productos. De cada proveedor se desea conocer su RUC, razón social y dirección.



1.3. TEMA 3 : REPRESENTACIÓN DE LA INFORMACIÓN

En Base de Datos existen muchos términos que seguro lo tenemos como saber previo tal es así que debemos distinguir entre los términos dato e información para poder entender los demás que usaremos en este curso.

Tomemos en cuenta que un **dato** es cualquier valor que puede representar algo, así como un número, imagen, sonido, símbolo, nombre, etc.; **su característica principal es que no permite tomar ninguna decisión**, ya que son simplemente datos que no tienen un significado concreto. Estos datos deben procesarse para dotarlos de significado y convertirlos en algo que nos pueda servir como **información**.

En cualquier proceso y construcción de un sistema informático, el diseño de la base de datos ocupa un lugar importante, a tal punto que esta puede verse como un proceso relativamente independiente dentro del diseño del sistema y compuesto por una serie de etapas. Es por ello que resulta de interés el estudio de los problemas relacionados con el diseño de las bases de datos y el modelamiento de la información.

1.3.1. Niveles de abstracción referidos a la información

Cuando se habla de información, se hace referencia, de forma general, a tres niveles diferentes de abstracción, tendiéndose a saltar de uno a otro sin establecer una advertencia previa.

1° NIVEL: MUNDO REAL

En este nivel encontramos a las entidades u objetos, que son elementos que componen nuestro universo y son considerados como elementos que existen y además están bien

diferenciados entre sí, estos poseen propiedades o características que son fácilmente detectables. **Por ejemplo:**

	Número de placa
	Color del auto
	Número de puertas
	Marca del auto
	Modelo del auto

Así mismo, podríamos nombrar a un teléfono móvil, alumno, empleado, producto, etc., pero no siempre será algo tangible, también podríamos nombrar a elementos que no son fácilmente palpables como un suceso o una transacción. **Por ejemplo:**

	Hora de inicio del siniestro
	Número de personas afectadas
	Distrito donde se ocasionó el siniestro
	Hora final del siniestro
	Número de bomberos

La determinación de cierta entidad u objeto correspondiente a un caso o proceso está muy relacionada con el nivel de abstracción en el que se esté realizando el análisis. Así, por ejemplo, si se estudia el comportamiento de un insecto específico en determinadas condiciones climáticas, las propiedades y relaciones que interesan son de un cierto tipo; sin embargo, si se estuviera realizando un estudio sobre las diferentes especies de insectos, serían otros los objetos por definir, así como las propiedades que los caracterizarían y las relaciones que se establecerían. Si se estuviera analizando todo el reino animal, serían también otros los objetos a definir, con sus características y propiedades.

2º NIVEL: DOMINIO DE LAS IDEAS

En este nivel se decide la información que debe existir en la base de datos sobre un caso o proceso del mundo real, es decir, qué debe almacenarse.

Aquí, es donde realmente se define el contenido informativo que representará al caso, proceso o ente de la realidad objetiva que se está analizando. De modo que, en este nivel, se definen qué objetos y propiedades son representativas, ya que sobre estos es necesario almacenar información. En este nivel, se trabaja con los conceptos más importantes del modelo de datos, que establecen la relación entre el mundo real y la información almacenada físicamente en la base de datos, como un dato o registro que definiremos a continuación:

- **Campo o Atributo:**

Es la unidad menor de información de un determinado objeto que se almacena en una base de datos y representa una propiedad de un objeto; por ejemplo, el color, tamaño o algo particular del objeto.

Sin embargo, hay que distinguir entre el nombre y el valor del atributo, ya que un nombre de atributo puede tomar diferentes valores sobre un cierto conjunto que se denomina dominio. A un valor de un atributo se le denomina ocurrencia del atributo. Por ejemplo:

	Atributo	Dominio	Ocurrencia
	Color	{Azul, Rojo, Gris}	Gris
	Marca	{Toyota, Kia, Chery}	Kia
	Modelo	{Rav4, Sportage, Tiggo}	Sportage

Ahora bien, una colección identificable de atributos es un registro y representa un objeto con sus propiedades. Una vez más, es imprescindible distinguir entre nombre y ocurrencia de artículo.

Una ocurrencia de artículo o tupla consiste en un grupo de ocurrencias de campos relacionados, representando una asociación entre ellos. Por ejemplo, tenemos un artículo correspondiente al objeto profesor, en un caso o proceso de la realidad que pretenda representar el comportamiento de una facultad. El nombre o tipo de artículo puede ser **PROFESOR**, que esté formado por los siguientes tipos de campos o atributos:

	Atributo	Descripción
	COD_PROF	Código único del profesor.
	NOM_PROF	Nombre completo del profesor.
	CAT_PROF	Categoría asignada al profesor.
	SUE_PROF	Sueldo asignado al profesor.

Una ocurrencia de este artículo puede ser:

PJQUISPE	JULIO CASTRO VERA	PARCIAL	S/. 3500.00
----------	-------------------	---------	-------------

▪ Archivo o archivos:

Son un conjunto de ocurrencias de un mismo tipo de artículo. En la práctica, llama la atención las colecciones o conjuntos de objetos similares. Además, es necesario almacenar la información de las mismas propiedades para cada uno de ellos; por ejemplo, el conjunto de profesores de la facultad.

Entonces, una base de datos contendrá muchas ocurrencias de cada uno de los tipos de artículos, lo que implica que la base de datos, por supuesto, también contendrá muchas ocurrencias de los distintos tipos de atributos.

Uno de los momentos cruciales en el diseño de un caso de la realidad objetiva que se concreta en una base de datos es, precisamente, la selección de los conjuntos de objetos y sus propiedades. Además, existe otro concepto muy importante en este nivel que es el concepto de **llave o clave**. Se denomina a este último como un atributo o conjunto de atributos de un objeto o entidad que define que cada ocurrencia de del objeto de la base de datos sea única. En principio, cada objeto o entidad tiene una llave, ya que se tiene como hipótesis que cada elemento u ocurrencia del objeto o entidad es diferente de las demás. Por ejemplo, el **código del profesor** puede constituir la llave de la entidad **profesor**.

3º NIVEL: DATOS

El tercer nivel corresponde a los datos propiamente dichos, los cuales son representados mediante cadenas de caracteres o de bits.

En este nivel es necesario tener en cuenta la diferencia entre tipo de dato y valor del dato. El tipo de dato corresponde a un atributo o tipo de atributo, que está asociado a

un tipo de artículo correspondiente, mientras que, el valor corresponde a una ocurrencia del atributo.

Sin embargo, una colección de bits o caracteres que representa un único valor de datos y que puede existir independientemente de cualquier información que se almacena adquiere significado solo cuando se le asocia a un tipo de atributo. Se puede, por ejemplo: almacenar permanentemente los valores ROJO, AZUL, VERDE, etc. y asociarlos en un momento determinado a un tipo de atributo a través de los valores que toma, representando una ocurrencia en una tupla. Veamos la integración de todos los niveles expresados en la entidad Auto:



1.3.2. Reglas de negocio

Toda organización funciona siguiendo múltiples reglas o también llamada políticas, estas pueden ser explícitas o tácitas las cuales se encuentran integradas en los procesos, aplicaciones informáticas, documentos, etc.

Así mismo, se podría decir que describe políticas, normas, operaciones, definiciones y restricciones presentes en una organización y que son de vital importancia para alcanzar los objetivos. Mencionaremos algunos ejemplos:

- ✓ Un docente dictará únicamente en una sede.
- ✓ Un cliente solo puede adquirir 3 promociones.
- ✓ Despues de 15' se considera tardanza.
- ✓ Por 5 minutos de tardanza equivale a una hora de descuento.

Por otra parte, una aplicación informática permite reflejar parte del funcionamiento del mundo real, haciendo que un proceso sea manejado por un usuario de manera correcta; para que esto ocurra debemos aplicar restricciones, de modo que se prevea acciones de manera correcta. Para un sistema de base de datos podríamos nombrar las siguientes reglas de negocio:

- ✓ Controlar el saldo negativo de un cliente.
- ✓ Controlar el stock negativo de un producto.
- ✓ Crear boletas de ventas a clientes que no se encuentran registrados.
- ✓ Controlar el número de intentos al ingresar un usuario y una clave.
- ✓ A los clientes de tipo A les aplicamos un descuento del 10% en pedidos superiores a 3.000.

1.3.3. Relaciones de correspondencia

Es importante notar que, en general, habrá asociaciones o relaciones enlazando las entidades básicas. Estos enlaces se pueden establecer entre diferentes objetos o tipos de artículos o entre un mismo tipo de artículo. **Por ejemplo:**



Se puede tener una relación entre dos tipos de objetos: **PROVEEDOR** y **PRODUCTO**, de modo que un proveedor puede suministrar muchos productos y que un producto puede ser suministrado por muchos proveedores y se conoce, además, la **CANTIDAD** de cada producto que suministra un proveedor dado.

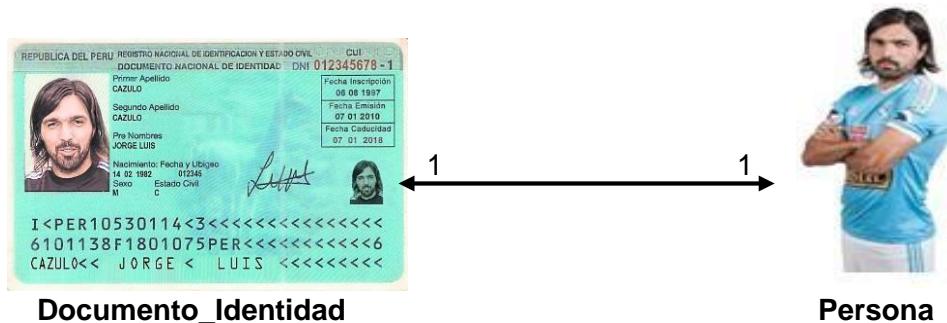
Es necesario profundizar acerca de los diferentes tipos de relaciones que pueden ocurrir en la práctica y establecer la correspondencia que existe entre los datos. Esta relación puede ser **simple** o **compleja**.

1.3.3.1. Relación simple

1:1 relación uno a uno

Se le llama así, a la relación cuya correspondencia es catalogada como biunívoca también llamada relación de **UNO A UNO**; entre las ocurrencias de las entidades.

Matemáticamente podríamos decir que se asocia cada uno de los elementos de un conjunto con uno, y solo uno, de los elementos de otro conjunto. **Así, por ejemplo:**



Las entidades **Documento_Identidad** y **Persona** se encuentran asociados, la correspondencia entre ellos es **simple**, por lo tanto, podríamos decir que: “*A cada persona le corresponde un documento de identidad y viceversa*”.

1.3.3.2. Relación compleja

1:N relación uno a muchos

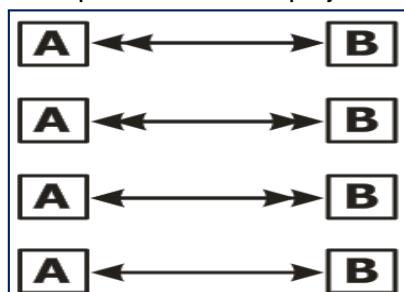
Se le llama así, a la relación cuya correspondencia es catalogada como unívoca y recíproca. Así, por ejemplo:



Si las entidades son **Curso** y **Estudiante**, la relación es más complicada, porque en cada curso, están inscritos varios alumnos. La terminología usual expresa que la correspondencia de **estudiante a curso** es simple, ya que cada estudiante es miembro de un único curso, mientras que la correspondencia de **curso a estudiante** es compleja, pues cada curso tiene, por lo general, muchos estudiantes.

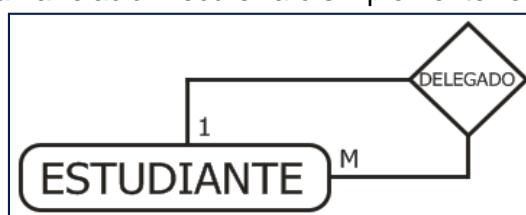
Finalmente, hay cuatro tipos de relaciones posibles entre dos tipos de artículos **A** y **B**:

- ✓ La correspondencia de A a B puede ser simple y la recíproca compleja.
- ✓ La correspondencia de A a B puede ser compleja y la recíproca simple.
- ✓ Ambas correspondencias pueden ser complejas o ambas pueden ser simples.



1.3.3.3. Características con respecto a las correspondencias

- Aunque la mayoría de las relaciones asocian dos tipos de entidades, éste no es siempre el caso. Por ejemplo, CURSO_HORARIO_ESTUDIANTE. Esto podría representar el hecho de que un curso se dicta a una cierta hora a un cierto estudiante. Esto no es lo mismo que la combinación CURSO_HORARIO y CURSO_ESTUDIANTE, ya que la información de que "el curso C1 se dicta en el horario H1 al estudiante E1" dice más que la combinación "el curso C1 se dicta en el horario H1" y "el estudiante E1 recibe clases en el horario H1".
- Las relaciones pueden establecerse entre un mismo tipo de entidad. Por ejemplo, una asociación entre un estudiante y otro puede venir dada por el hecho de que un estudiante sea el delegado de otros estudiantes. A este tipo de relación se le llama relación recursiva o simplemente recursividad.



- Es importante señalar que una asociación entre entidades puede ser considerada en sí como una entidad, ya que una relación es concebida como un objeto bien diferenciado sobre el cual se desea almacenar información.
- Entonces, un modelo de datos no es más que la representación de un caso de la realidad objetiva a través de los objetos, sus propiedades y las relaciones que se establecen entre ellos.

1.3.4. Ejercicios de aplicación

Caso: BIBLIOTECA

En una biblioteca, se desea diseñar la base de datos para el control de los préstamos de libros. De cada libro se conoce el código que lo identifica, su título y la cantidad de páginas que tiene.

Así mismo, un **libro** se clasifica por una materia y por ésta se clasifican muchos libros. De cada **materia** se conoce el código que la identifica y su nombre. Los libros tienen muchos **ejemplares**, pero un ejemplar lo es de un solo libro. De cada ejemplar se sabe su código y su estado de conservación.

Finalmente, un libro se les puede prestar a muchos **usuarios** y a un usuario se le puede prestar muchos ejemplares; del usuario se conoce su DNI, nombre y apellido paterno, su dirección, y su ocupación.

Del caso expuesto, debe tener en cuenta:

- ✓ Determine las entidades y lístelos.
- ✓ Determine los atributos por cada entidad encontrada.
- ✓ Especifique el nivel de datos por cada entidad.
- ✓ Especifique las relaciones de correspondencia entre las entidades.

Solución:

- ✓ Determinando las entidades del caso



- ✓ Determinando los atributos por cada entidad

Listado de entidades	Listado de atributos
LIBRO	Código que lo identifica, título y la cantidad de páginas que tiene.
MATERIA	Código que lo identifica y nombre de la materia.
EJEMPLAR	Código que lo identifica y estado de conservación.
USUARIO	Número de DNI, nombres, apellido paterno, dirección y ocupación.

✓ Especificando el nivel de datos del caso

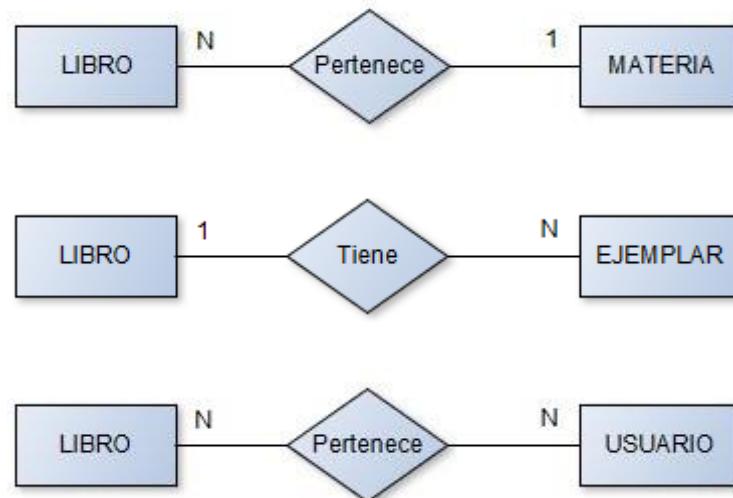
Código	Titulo	Cantidad de Paginas
L001	BASE DE DATOS RELACIONALES	500
L002	VISUAL C# 2014	600

Código	Nombre
M001	BASE DE DATOS
M002	LENGUAJE DE PROGRAMACIÓN

Código	Estado de conservación
E001	BUENO
E002	REGULAR

Código	Nombres	Apellido Paterno	Dirección	Ocupación
U001	MARIA	ACOSTA	AV. EL SOL 657	DOCTOR
U002	GUADALUPE	ZAMORA	AV. LAS FLORES 3458	ABOGADO

✓ Especificando las relaciones de correspondencia



1.3.5. Autoaprendizaje

- Defina los conceptos de base de datos y de sistema de gestión de bases de datos.

2. Mencione 10 entidades que usted conozca. Defina como máximo 5 atributos para cada una y decida cuál puede ser la llave primaria en cada entidad.

Nro.	Entidad	Atributo1	Atributo2	Atributo3	Atributo4	Atributo5
1	Estudiante	Código (PK)	Nombre	Dirección	Teléfono	Fecnac
2						
3						
4						
5						
6						
7						
8						
9						
10						

3. Mencione 10 relaciones que sea capaz de reconocer. Defina el tipo de éstas y algunos atributos para cada una siempre que sea posible. (Identifique: entidades, tipo de relación, cardinalidad, atributos).
4. Aplique la terminología explicada en clases, correspondiente al nivel del dominio de las ideas, y describa las relaciones de correspondencia existentes en los siguientes casos de la realidad objetiva. Para ello, represente en un gráfico, de acuerdo con su criterio, las entidades y sus atributos, indicando los atributos que constituyen las llaves primarias, las relaciones y el tipo de éstas, así como sus atributos.
5. Para la solución de los siguientes casos, deben tener en cuenta:
- ✓ Determine las entidades y lístelas.
 - ✓ Determine los atributos por cada entidad encontrada.
 - ✓ Especifique el nivel de datos por cada entidad.
 - ✓ Especifique las relaciones de correspondencia entre las entidades.

CASO 1:

Para el control de las escuelas primarias en la ciudad de Lima se tiene la siguiente información:

De cada escuela, un número que la identifica, su nombre y su dirección. De cada aula, el número que la identifica, la cantidad de asientos que tiene y el piso en que se encuentra situada.

De cada grupo de clases, un identificador del grupo, el grado escolar del grupo y la cantidad de alumnos que tiene. De cada maestro, su DNI, su nombre, su sexo y el año en que se graduó. De cada alumno, su número de expediente, su nombre, su sexo y su fecha de nacimiento. Una escuela tiene muchas aulas y muchos grupos de clases, pero cada aula pertenece a una escuela y lo mismo sucede con cada grupo. Un grupo siempre recibe clases en la misma aula y un aula pertenece a un solo grupo. En un grupo imparte clases un maestro y éste sólo imparte clases en un grupo. En cada grupo de clases hay muchos alumnos, pero un alumno forma parte de un solo

CASO 2:

En un centro de información científica (biblioteca), se desea controlar el uso de la bibliografía que existe. En el centro existen varias salas. De cada sala se conoce el número que la identifica, especialidad y cantidad de empleados.

En cada sala están disponibles revistas y libros. De cada revista se conoce el código que la identifica, nombre, fecha de publicación y país de procedencia. De cada libro se conoce su código, título, editorial y país de procedencia. Cada libro y revista existente sólo se encuentra en una sala. Las revistas sólo pueden ser consultadas en las salas; sin embargo, los libros pueden ser solicitados en préstamo por los usuarios, llevándose el control de dichos préstamos. Un libro puede ser prestado a varios usuarios (durante la existencia del libro) y un usuario puede solicitar varios libros. De cada usuario se sabe su DNI, nombre, distrito en que reside y departamento en que trabaja. Para cada libro se conoce la fecha de inicio de un préstamo realizado a determinado usuario.

CASO 3:

En los centros de beneficio agrícola se les aplica diversos tratamientos a los tubérculos.

De cada tipo de tubérculo (papa, yuca, camote, etc.), se conoce el código que lo identifica, su descripción y la variedad a la que pertenece. Cada tipo de tubérculo puede ser enviado a distintos centros de beneficio y en un centro de beneficio pueden ser tratados distintos tipos de tubérculos. De cada centro de beneficio se sabe el código que lo identifica, su dirección y la cantidad de trabajadores que tiene. A cada centro de beneficio se le asigna el plan (en kilogramos) de atención a un tipo de tubérculo.

En un centro de beneficio se aplican diferentes tratamientos y un tratamiento puede ser aplicado en diferentes centros de beneficio. De cada tratamiento, se conoce el código que lo identifica, su nombre y el objetivo que persigue.

CASO 4:

En un hospital se desea controlar la actividad asistencial que se brinda en las consultas de la Sala de Emergencias.

Allí brindan sus servicios los médicos organizados en equipos. A un equipo pertenecen varios médicos y un médico pertenece a un equipo. De cada equipo se conoce el código que lo identifica, el nombre del jefe del equipo y la periodicidad con la que le corresponde hacer guardia al equipo. De cada médico se conoce su DNI, nombre, especialidad y categoría. Los pacientes que llegan al Cuerpo de Guardia pueden ser atendidos por varios médicos (si sus síntomas indican la necesidad de que varios especialistas lo asistan) y un médico atiende a muchos pacientes. De cada paciente se sabe su DNI, nombre, edad, sexo y ocupación. Se sabe el tiempo dedicado por el médico a la atención de un determinado paciente, así como el diagnóstico que le hizo y el tratamiento que le indicó.

Los médicos en la Sala de Emergencias pueden utilizar en su labor asistencial diferentes medios de diagnóstico (Rayos X, análisis, etc.) y un medio de diagnóstico puede ser empleado por muchos médicos. De cada medio de diagnóstico se conoce el código que lo identifica, su descripción y el costo por unidad. Para cada médico se conoce la cantidad de veces que ha ordenado la aplicación de un medio de diagnóstico dado.

1.4. TEMA 4: ARQUITECTURA DE UN SISTEMA DE BASE DE DATOS



Figura 2: Vistas de una base de datos

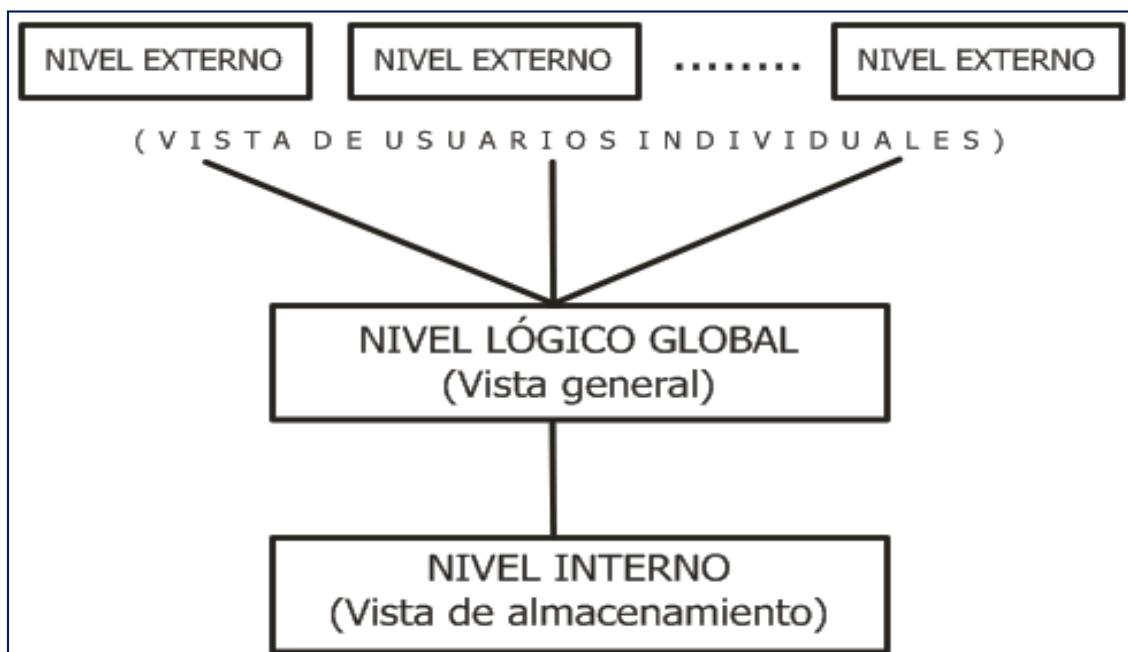
Fuente.- Tomado de http://4.bp.blogspot.com/_IlyXiNJht4/TQpziVPyon/AAAAAAAABU/legf33kEanE/s320/vistas.png

1.4.1. Arquitectura de un sistema de base de datos

A continuación, presentaremos la arquitectura de un **SGBD**, aunque no podemos asegurar que cualquier **SGBD** se corresponda exactamente con ella. Sin embargo, esta arquitectura corresponde suficientemente bien con un gran número de sistemas. Además, está de acuerdo con la arquitectura propuesta por el grupo **ANSI/SPARC** (American National Standards Institute, Standards Planning And Requirements Committee).

La arquitectura se divide en tres niveles generales:

- El **nivel interno** es el más cercano al almacenamiento físico, o sea, es el relacionado con la forma en que los datos están realmente almacenados.
- El **nivel externo** es el más cercano a los usuarios, o sea, es el relacionado con la forma en que los datos son vistos por cada usuario individualmente.
- El **nivel lógico global** es un nivel intermedio entre los dos anteriores.



Existirán varias "vistas externas" diferentes. Cada vista es una representación más o menos abstracta de alguna porción de la base de datos total y existirá únicamente una "vista general", consistente en una representación también abstracta de la base de datos

en su totalidad. Igualmente, existirá una única "vista interna" que representa a la base de datos completa, tal y como está realmente almacenada.

1.4.1.1. Nivel Externo

Es el nivel del usuario individual, donde un usuario puede ser un programador de aplicación o un usuario final con cualquier grado de sofisticación. Cada usuario tiene un lenguaje a su disposición:

- Para el programador, ese lenguaje puede ser un lenguaje de programación convencional, tal como Java, o un lenguaje de programación específico de un sistema, tal como el Visual C#.
- Para el usuario final, el lenguaje puede ser un lenguaje de consulta (interrogaciones, query) o un lenguaje de propósito especial, quizás basado en sistemas de menús y ventanas, y construido para satisfacer los requerimientos de un usuario, y se encuentra soportado por algún programa de aplicación en línea.

Es importante señalar que todo lenguaje debe incluir un sublenguaje de datos, o sea, un subconjunto del lenguaje que trata específicamente con los objetos de la base de datos y sus operaciones.

En principio, cualquier sublenguaje de datos es realmente una combinación de, al menos, dos lenguajes subordinados: un **lenguaje de definición de datos (DDL)**, el cual garantiza la definición o descripción de los objetos de la base de datos, y un **lenguaje de manipulación de datos (DML)**, el que garantiza la manipulación o procesamiento de esos objetos.

Ya se ha indicado que un usuario individual estará, generalmente, interesado solo en cierta porción de la base de datos. Aún más, la vista de esa porción será generalmente abstracta cuando se compara con la forma en que los datos están físicamente almacenados. El término definido por el comité **ANSI/SPARC** para una vista de un usuario es **vista externa**, la cual es el contenido de la base de datos tal y como es vista por un usuario en particular. Dicho de otra manera, para ese usuario, la vista externa es la base de datos.

El sublenguaje de datos del usuario se define en términos de artículos externos; por ejemplo, una operación del DML que permite recuperar artículos, generará una ocurrencia de artículos externos y no una ocurrencia de artículos almacenados.

Cada vista externa se define mediante un esquema externo consistente, básicamente, en definiciones de cada uno de los diferentes tipos de artículos externos en esa vista. El esquema externo se escribe usando la porción del **DDL** del sublenguaje de datos del usuario; además, tiene que existir una definición de la correspondencia entre el esquema externo y el esquema lógico global.

1.4.1.2 Nivel Lógico Global

La vista lógica es una representación del contenido informativo total de la base de datos. En comparación con la forma en que los datos están almacenados físicamente, es una forma **abstracta**.

Esta vista puede ser muy diferente de la forma en la que los datos son vistos por un usuario en particular. Pretende ser una vista de los datos tal como son en lugar de como

los usuarios están forzados a verlos por las restricciones, digamos, de un lenguaje particular o de un determinado hardware que utilicen.

Consiste en múltiples ocurrencias de múltiples tipos de artículos lógicos. Por ejemplo, puede ser una colección de ocurrencias de artículos de departamentos, más una colección de ocurrencia de artículos de empleados, etc. Un artículo lógico no es necesariamente igual a un artículo externo ni a un artículo almacenado. La vista lógica se define mediante el esquema lógico que incluye las definiciones de cada uno de los diferentes tipos de artículos lógicos. El esquema lógico se describe usando otro lenguaje de definición de datos: el **DDL** lógico. Si se desea lograr la independencia de los datos, entonces las definiciones del **DDL** lógico no deben comprender ninguna consideración sobre la estructura de almacenamiento ni la estrategia de acceso. Ellas tienen que ser definiciones sólo referentes al contenido informativo.

Entonces, es una vista del contenido total de la base de datos y el esquema lógico es una definición de esa vista. Sin embargo, el esquema lógico no es, simplemente, un conjunto de definiciones como las que se encuentran, por ejemplo, en un programa Java. Las definiciones en el esquema lógico deben incluir una gran cantidad de aspectos adicionales, tales como los chequeos de protección y los chequeos de integridad. En la mayoría de los sistemas actuales, el esquema lógico es realmente sólo un poco más que la simple unión de todos los esquemas externos individuales; posiblemente con la adición de algunos chequeos simples de protección e integridad. Sin embargo, está claro que los sistemas del futuro soportarán un nivel lógico mucho más sofisticado, que permita también describir la forma en que se usan los datos, cómo fluyen de un punto a otro, para qué se usan en cada punto, a qué controles son sometidos, etc.

1.4.1.3 Nivel Interno

La vista interna es una representación de bajo nivel de la base de datos completa, que consiste en múltiples ocurrencias de múltiples tipos de artículos internos.

"Artículo interno" es el término definido por **ANSI/SPARC** para la construcción que hasta ahora hemos llamado artículo almacenado. La vista interna está entonces aún a un paso del nivel físico, ya que ella no opera en términos de artículos físicos (también llamados páginas o bloques) ni con consideraciones específicas de los equipos, tales como tamaños de sectores o pistas.

Básicamente, la vista interna asume un espacio de dirección lineal infinita. Los detalles de cómo se hace corresponder ese espacio con el almacenamiento físico son muy específicos de un sistema y deliberadamente se omitieron de la arquitectura.

La vista interna se describe mediante el esquema interno, el cual no sólo define los diferentes tipos de artículos almacenados, sino que también especifica los índices que existen, la representación de los campos almacenados, la secuencia física en que están los artículos almacenados, etc. El esquema interno se describe usando otro lenguaje de definición de datos: el **DDL** interno.

1.4.1.4 Correspondencias entre los niveles de arquitectura

En el esquema presentado de la arquitectura de un **SGBD**, se observan los niveles de correspondencias, uno entre los niveles externo y lógico global y otro entre los niveles lógico global e interno.

- La **correspondencia lógica/interna** especifica la forma en que los artículos y campos lógicos se representan en el nivel interno. Además, si se cambia la estructura de la vista interna, mejor dicho, si se hace un cambio en el esquema interno, entonces, la correspondencia lógica/interna tiene también que cambiar, de modo que el esquema lógico permanezca invariable. En otras palabras, los efectos de estos cambios deben ser aislados por debajo del nivel lógico para que se mantenga la independencia de los datos.
- Existe también una **correspondencia externa/lógica** entre cada vista externa particular y la vista lógica. Las diferencias que pueden existir entre estos dos niveles son similares a las que pueden existir entre las vistas lógica e interna. Por ejemplo, los campos pueden tener diferentes tipos de datos, se pueden cambiar los nombres de artículos y campos, múltiples campos lógicos pueden ser combinados en un único campo externo, etc. Puede existir al mismo tiempo cualquier cantidad de vistas externas; cualquier cantidad de usuarios puede compartir una vista externa dada; las diferentes vistas externas se pueden solapar. Algunos sistemas permiten la definición de una vista externa a partir de otra (mediante una correspondencia externa/externa); esta característica es útil cuando varias vistas externas están estrechamente relacionadas entre sí.

1.4.2. Organizaciones de archivos y el nivel interno de la arquitectura

Físicamente, se puede decir que las bases de datos se almacenan siguiendo diferentes organizaciones de archivos. Cada una de estas tiene distintas características de desempeño; ninguna se puede decir que sea óptima para todas las aplicaciones, sino que se decide emplear una u otra en dependencia de la aplicación.

A continuación, presentaremos las principales características de las más usadas organizaciones de archivos, de modo que puedan tener ideas de cómo se pueden organizar los datos en estos archivos para lograr determinados objetivos. Pero, antes de ello, abordaremos algunas definiciones y características relativas a los archivos.

Empezaremos diciendo que la utilización de los archivos se debe a dos causas fundamentales:

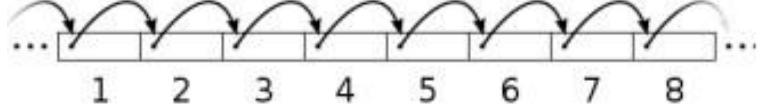
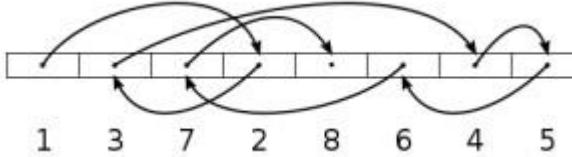
- Manejo de mucha información
- Almacenamiento de información permanente

Ya que el trabajo en memoria interna es muy rápido, pero es un recurso relativamente escaso y caro, y la memoria periférica es más barata, aunque más lenta. Además, influye en esto el hecho de que el contenido de la memoria central se pierde al cesar el fluido eléctrico, lo que hace aconsejable almacenar en soportes externos los grandes volúmenes de información.

1.4.2.1 Operaciones fundamentales sobre archivos

Las operaciones fundamentales que se realizan sobre archivos pueden colocarse en dos grandes grupos:

Acceso	Se entiende el acceso a un archivo como la forma por la cual es posible tener conocimiento de la información contenida en los respectivos registros, mejor dicho, el modo como se pueden leer los registros del archivo, las ocurrencias de estos registros almacenadas en él.
---------------	--

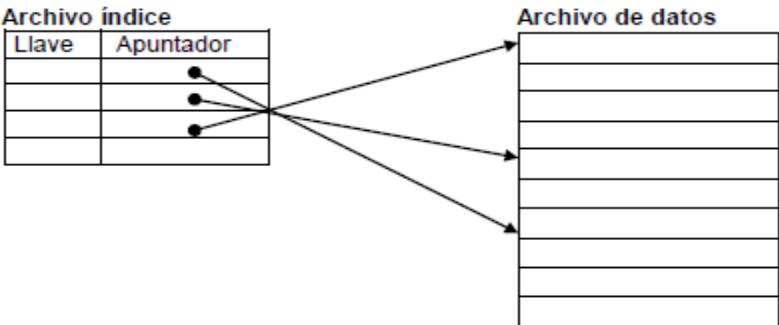
	<p>Las diferentes formas de accesar a un archivo pueden agruparse en dos tipos:</p> <ol style="list-style-type: none"> En secuencia o secuencial. Se caracteriza por el hecho de que, en la lectura del archivo, es obligatorio que, a continuación del tratamiento del registro de orden n, se traten los de orden $n+1$, $n+2$, ... hasta un límite determinado. Puede realizarse desde el inicio del archivo o a partir de un registro n.  <ol style="list-style-type: none"> Aleatorio o directo. Se caracteriza porque se lee, y consecuentemente trata cualquier registro en cualquier orden (a través de la llave o una transformación de ésta). No puede realizarse en periféricos de acceso secuencial. 
Actualización	<p>La mayoría de los archivos deben ser actualizados con el transcurso del tiempo para que se ajusten a la realidad. La actualización de un archivo se realiza mediante tres operaciones fundamentales:</p> <ul style="list-style-type: none"> Alta: Se le llama así a la creación de un nuevo registro en el archivo. Baja: Se le llama así cuando se elimina un registro. Esta puede ser lógica o física. Modificación: Se le llama así cuando se cambia alguna información en el registro. <p>Para realizar las actualizaciones es importante validar los datos que se van a actualizar.</p>

1.4.2.2 Organización de los archivos

Es necesario indicar que la bibliografía existente al respecto asume diferentes posiciones respecto a la terminología de emplear. En muchos casos, las definiciones son ambiguas y hay poco acuerdo entre los textos que tratan este tema.

A pesar de esto, en la actualidad, las organizaciones generalmente aceptadas son:

Organización Secuencial	Es la forma más sencilla para almacenar los registros de un archivo, uno después de otro, a continuación del otro. Todos los registros se
--------------------------------	---

	<p>almacenar por su posición: uno es el primero, el siguiente es el segundo y así sucesivamente.</p> <p>Es la más vieja forma de organizar un archivo y fue empleada para las cintas magnéticas desde los inicios de la computación.</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <tr><td>1</td><td>2</td><td>...</td><td>n</td><td>n+1</td><td>...</td></tr> </table> <p>Así, por ejemplo, el archivo POSTULANTE, en el que están las ocurrencias de postulantes una a continuación de otra; sería de la siguiente manera:</p> <table border="1" style="width: 100%; border-collapse: collapse;"> <thead> <tr> <th>Nº</th><th>DNI</th><th>NOMBRES</th><th>FECHA</th><th>HORA</th></tr> </thead> <tbody> <tr><td>1</td><td>43462698</td><td>ABANTO JIMES ANGELA MILAGROS</td><td>07/05/2012</td><td>09:00</td></tr> <tr><td>2</td><td>45017141</td><td>AVALOS BARAHONA LESDY TANIA</td><td>07/05/2012</td><td>09:00</td></tr> <tr><td>3</td><td>41194541</td><td>AVELLANEDA BAUTISTA SUHGEY MAVILA</td><td>07/05/2012</td><td>09:00</td></tr> <tr><td>4</td><td>10798934</td><td>BOCANEGRA ARIAS SONIA KAREN</td><td>07/05/2012</td><td>09:20</td></tr> <tr><td>5</td><td>43524781</td><td>CALDERON VALERA DORIS HAYDEE</td><td>07/05/2012</td><td>09:20</td></tr> <tr><td>...</td><td>...</td><td>...</td><td>...</td><td>...</td></tr> <tr><td>12</td><td>43129691</td><td>GARCIA CRISTOBAL CARLOS ALOR</td><td>07/05/2012</td><td>10:00</td></tr> <tr><td>13</td><td>42626923</td><td>HUANACUNI TICONA SOLEDAD</td><td>07/05/2012</td><td>10:20</td></tr> <tr><td>14</td><td>31037402</td><td>HUAYHUAS ROJAS PITHER</td><td>07/05/2012</td><td>10:20</td></tr> <tr><td>15</td><td>04080376</td><td>ILLANES BUSTAMANTE JOSE LUIS</td><td>07/05/2012</td><td>10:20</td></tr> <tr><td>16</td><td>09019417</td><td>LA MATTA CASTRO LUIS ALBERTO</td><td>07/05/2012</td><td>10:40</td></tr> </tbody> </table>	1	2	...	n	n+1	...	Nº	DNI	NOMBRES	FECHA	HORA	1	43462698	ABANTO JIMES ANGELA MILAGROS	07/05/2012	09:00	2	45017141	AVALOS BARAHONA LESDY TANIA	07/05/2012	09:00	3	41194541	AVELLANEDA BAUTISTA SUHGEY MAVILA	07/05/2012	09:00	4	10798934	BOCANEGRA ARIAS SONIA KAREN	07/05/2012	09:20	5	43524781	CALDERON VALERA DORIS HAYDEE	07/05/2012	09:20	12	43129691	GARCIA CRISTOBAL CARLOS ALOR	07/05/2012	10:00	13	42626923	HUANACUNI TICONA SOLEDAD	07/05/2012	10:20	14	31037402	HUAYHUAS ROJAS PITHER	07/05/2012	10:20	15	04080376	ILLANES BUSTAMANTE JOSE LUIS	07/05/2012	10:20	16	09019417	LA MATTA CASTRO LUIS ALBERTO	07/05/2012	10:40
1	2	...	n	n+1	...																																																														
Nº	DNI	NOMBRES	FECHA	HORA																																																															
1	43462698	ABANTO JIMES ANGELA MILAGROS	07/05/2012	09:00																																																															
2	45017141	AVALOS BARAHONA LESDY TANIA	07/05/2012	09:00																																																															
3	41194541	AVELLANEDA BAUTISTA SUHGEY MAVILA	07/05/2012	09:00																																																															
4	10798934	BOCANEGRA ARIAS SONIA KAREN	07/05/2012	09:20																																																															
5	43524781	CALDERON VALERA DORIS HAYDEE	07/05/2012	09:20																																																															
...																																																															
12	43129691	GARCIA CRISTOBAL CARLOS ALOR	07/05/2012	10:00																																																															
13	42626923	HUANACUNI TICONA SOLEDAD	07/05/2012	10:20																																																															
14	31037402	HUAYHUAS ROJAS PITHER	07/05/2012	10:20																																																															
15	04080376	ILLANES BUSTAMANTE JOSE LUIS	07/05/2012	10:20																																																															
16	09019417	LA MATTA CASTRO LUIS ALBERTO	07/05/2012	10:40																																																															
Organización Indizada	<p>Los accesos a los registros almacenados se realizan a través de un índice. La forma básica de un índice incluye una llave de registro y la dirección de almacenamiento para éste. Para encontrar un registro específico, se rastrea primero el índice y, al encontrar la dirección, se accede directamente el registro.</p>  <p>Entonces, un índice es un archivo auxiliar que se utiliza para accesar los registros de otro archivo, que llamaremos archivo principal o de datos, por el valor de un dato o conjunto de datos, que es la llave o clave de indización.</p> <p>A los registros del índice se les llama entradas. Cada entrada corresponde a un valor o intervalo de valores de la llave y es el padre cuyos hijos son los registros del archivo principal, en los</p>																																																																		

	<p>cuales la llave toma el valor, o un valor del intervalo de valores que corresponde a la entrada en cuestión.</p> <p>En la siguiente imagen se muestra un archivo índice muy simple en el que se tiene la llave y un puntero al dato en el archivo principal.</p> <table border="1"> <thead> <tr> <th>Llave</th><th>apuntador</th></tr> </thead> <tbody> <tr> <td>Álvarez, Luis</td><td>●</td></tr> <tr> <td>García, Berta</td><td>●</td></tr> <tr> <td>Lazo, Juan</td><td>●</td></tr> <tr> <td>López, María</td><td>●</td></tr> <tr> <td>Martínez, José</td><td>●</td></tr> </tbody> </table> <p>Archivo Índice</p> <table border="1"> <thead> <tr> <th>López, María</th><th>Técn A</th><th>...</th></tr> </thead> <tbody> <tr> <td>Lazo, Juan</td><td>Oper B</td><td>...</td></tr> <tr> <td>García, Berta</td><td>Oper C</td><td>...</td></tr> <tr> <td>Martínez, José</td><td>Técn A</td><td>...</td></tr> <tr> <td>Álvarez, Luis</td><td>Técn B</td><td>...</td></tr> </tbody> </table> <p>Archivo de datos</p>	Llave	apuntador	Álvarez, Luis	●	García, Berta	●	Lazo, Juan	●	López, María	●	Martínez, José	●	López, María	Técn A	...	Lazo, Juan	Oper B	...	García, Berta	Oper C	...	Martínez, José	Técn A	...	Álvarez, Luis	Técn B	...
Llave	apuntador																											
Álvarez, Luis	●																											
García, Berta	●																											
Lazo, Juan	●																											
López, María	●																											
Martínez, José	●																											
López, María	Técn A	...																										
Lazo, Juan	Oper B	...																										
García, Berta	Oper C	...																										
Martínez, José	Técn A	...																										
Álvarez, Luis	Técn B	...																										

1.4.3. Autoaprendizaje

- Mencione los niveles en que se divide la arquitectura de un SBD y las características fundamentales de cada uno de ellos.
- Explique cómo se establece la correspondencia entre los distintos niveles de la arquitectura de un SBD y cómo se logra garantizar la independencia de los datos respecto a los programas de aplicación.
- Mencione tres de las funciones principales que realiza un administrador de bases de datos.
- Explique cuáles son las principales operaciones que se realizan sobre los archivos y los distintos tipos de cada una de ellas.
- Describa, a grandes rasgos, los tres tipos de organización de archivos más empleados.
- Defina un archivo índice. ¿Cuáles son las estructuras de archivo índice que conoce? Explique las características de cada una.
- ¿Cuál es la relación del nivel interno de la arquitectura de un SBD con los archivos y las organizaciones de éstos?

Resumen

- ❑ El procesamiento automatizado de datos ha pasado por diferentes etapas en su desarrollo hasta llegar a la actual, en la que se emplean bases de datos para el almacenamiento de la información.
- ❑ El principal objetivo de los SBD es garantizar la independencia de los datos respecto a los programas de aplicación.
- ❑ Otro objetivo muy importante de los SBD es la minimización de la redundancia.
- ❑ **Entidad:** objeto del cual se describen ciertas características.
- ❑ **Campo o atributo:** es la unidad menor de información sobre un objeto (almacenada en la base de datos) y representa una propiedad de un objeto.
- ❑ Un atributo puede tomar diferentes valores sobre un cierto conjunto que se denomina **dominio**.
- ❑ **Dominio:** rango de valores posibles de un atributo.
- ❑ A un valor de un atributo definido en el dominio dado, en un cierto momento del tiempo, se le denomina **ocurrencia** del atributo.
- ❑ Un **artículo o registro** es una colección identificable de campos asociados y representa un objeto con sus propiedades.
- ❑ Una **ocurrencia de artículo o tupla** consiste en un grupo de ocurrencias de campos relacionados, representando una asociación entre ellos.
- ❑ Un **archivo o archivos** es un conjunto de ocurrencias de un mismo tipo de artículo.
- ❑ Una **base de datos** está formada por múltiples archivos.
- ❑ Existen asociaciones o **relaciones** enlazando las entidades, que pueden tener o no atributos. Pueden establecerse sobre la misma entidad o sobre entidades diferentes. En una relación puede participar cualquier cantidad de entidades.
- ❑ Las relaciones pueden ser de uno a uno (**1: 1**), de uno a muchos (**1: m**) y de muchos a muchos (**m: m**).
- ❑ La arquitectura de un SBD está compuesta por tres niveles: externo, lógico global e interno. El nivel lógico global de los datos representa el contenido informativo total de la base de datos y, desde el punto de vista del diseñador, es el más importante.
- ❑ Existe una correspondencia entre el nivel externo y el lógico global; y entre el lógico global y el interno. Para garantizar la independencia de los datos respecto a los programas de aplicación, cualquier cambio en el nivel interno debe reflejarse adecuadamente en la correspondencia interna/lógico global, para no afectar el nivel lógico global.
- ❑ Las principales operaciones que se realizan sobre los archivos son los siguientes: acceso, que puede ser secuencial o directo, y actualización, que se refiere a las altas, bajas y modificaciones de la información.
- ❑ El nivel interno de la arquitectura de un SBD puede considerarse estructurada por archivos con distintos modos de organización.
- ❑ Si desea saber más acerca de estos temas, puede consultar las siguientes páginas.

Link	Descripción
https://prezi.com/blhkclirsv3h/arquitectura-de-los-sistemas-de-bases-de-datos/	En esta página web hallará algunos conceptos complementarios a los mostrados en el manual sobre la arquitectura de una base de datos.

http://dataoteca.unad.edu.co/contenidos/301309/E xe- Modulo/MDS301309/mtodos_de_organizacin_de archivos.html	En esta página web encontrará definiciones complementarias al almacenamiento secuencial indexado.
http://es.ccm.net/contents/66-introduccion-bases-de-datos	En esta página web hallará algunos conceptos complementarios a los mostrados en el manual sobre la introducción a la base de datos.
http://elies.rediris.es/elies9/4-1-2.htm	En esta página web encontrará definiciones complementarias sobre las características y los objetivos de la base de datos.
http://es.slideshare.net/erickrwk/semana-1-t-sistema-de-base-de-datos	En esta página web encontrará ejercicios sobre la representación de la información.



MODELO CONCEPTUAL

LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la segunda unidad, el estudiante diseña el diagrama entidad relación (DER) de un proceso de negocio a partir de casos planteados por el profesor, relacionando las características del modelo conceptual con el diseño de una base de datos.

TEMARIO

2.1 Tema 5	: Modelo conceptual
2.1.1	: Modelo conceptual de datos
2.2 Tema 6	: Diagrama Entidad-Relación
2.2.1	: Entidades: Generalización / Especialización
2.2.2	: Entidad fuerte/débil
2.2.3	: Ejercicios de aplicación
2.2.4	: Relaciones recursivas
2.2.5	: Agregaciones
2.2.6	: Ejercicio integrador
2.2.7	: Autoaprendizaje

ACTIVIDADES PROPUESTAS

1. Explicaciones por parte del facilitador combinadas con ejemplos.
2. Se efectúan preguntas de comprobación dirigidas a los estudiantes sobre los temas tratados.
3. Se plantea un caso para identificar los componentes de un diagrama entidad-relación y se construye el modelo correspondiente.

2.1. TEMA 5: MODELO CONCEPTUAL

2.1.1. Modelo conceptual de datos

Características del modelo conceptual

El proceso de diseño de la base de datos transita a través de una serie de pasos en los cuales se va avanzando de un nivel de abstracción menor a otro más profundo, mediante la elaboración de una sucesión de modelos. En los últimos años, se ha generalizado la concepción del diseño de las bases de datos propuestas por el grupo **ANSI/SPARC**, la cual constituye, al mismo tiempo, una arquitectura para los **SBD**, tal y como la acabamos de estudiar.

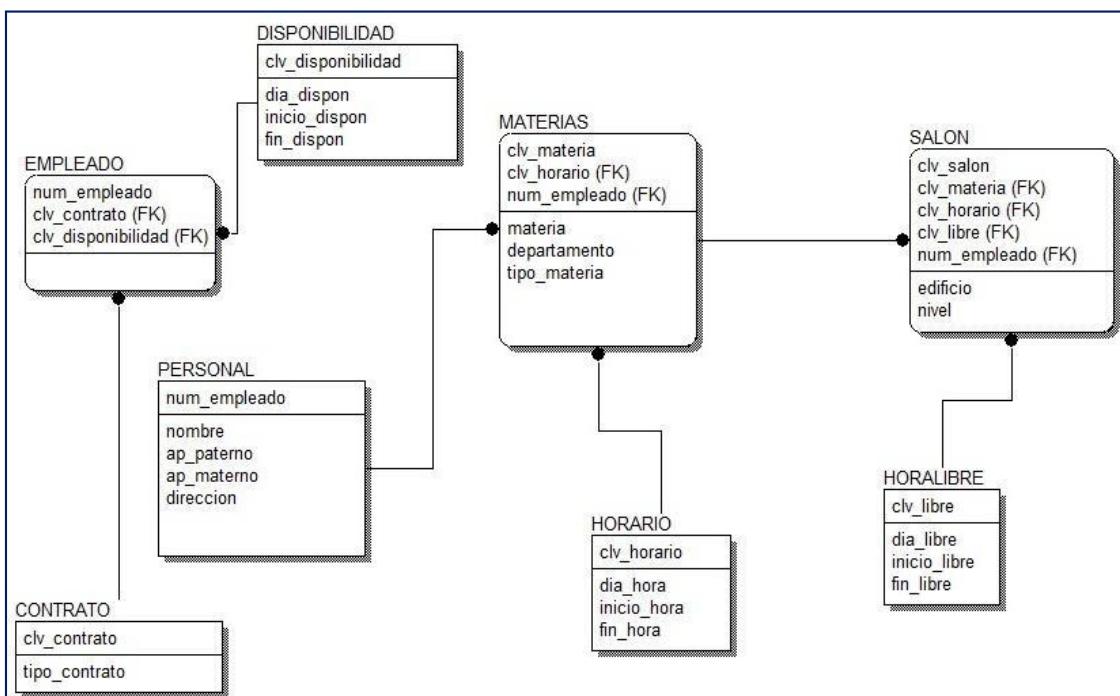


Figura 3: Modelo conceptual
Fuente. - Tomado de la aplicación Erwin

Hemos visto en esta arquitectura que cada nivel de la misma es una cierta forma de representación abstracta de la información y una de las funciones más importantes del **SGBD** consiste precisamente en permitirle al usuario la interacción con los datos en estos términos abstractos, en lugar de tenerlo que hacer directamente con la forma en que esos datos están físicamente almacenados. Es por ello que, al acometerse la tarea de diseño de una base de datos, la atención se debe centrar en el aspecto lógico de la información, ya que los detalles relacionados con el almacenamiento físico son parte de todo **SGBD** comercial que se utilice y, por tanto, no pueden ser modificados.

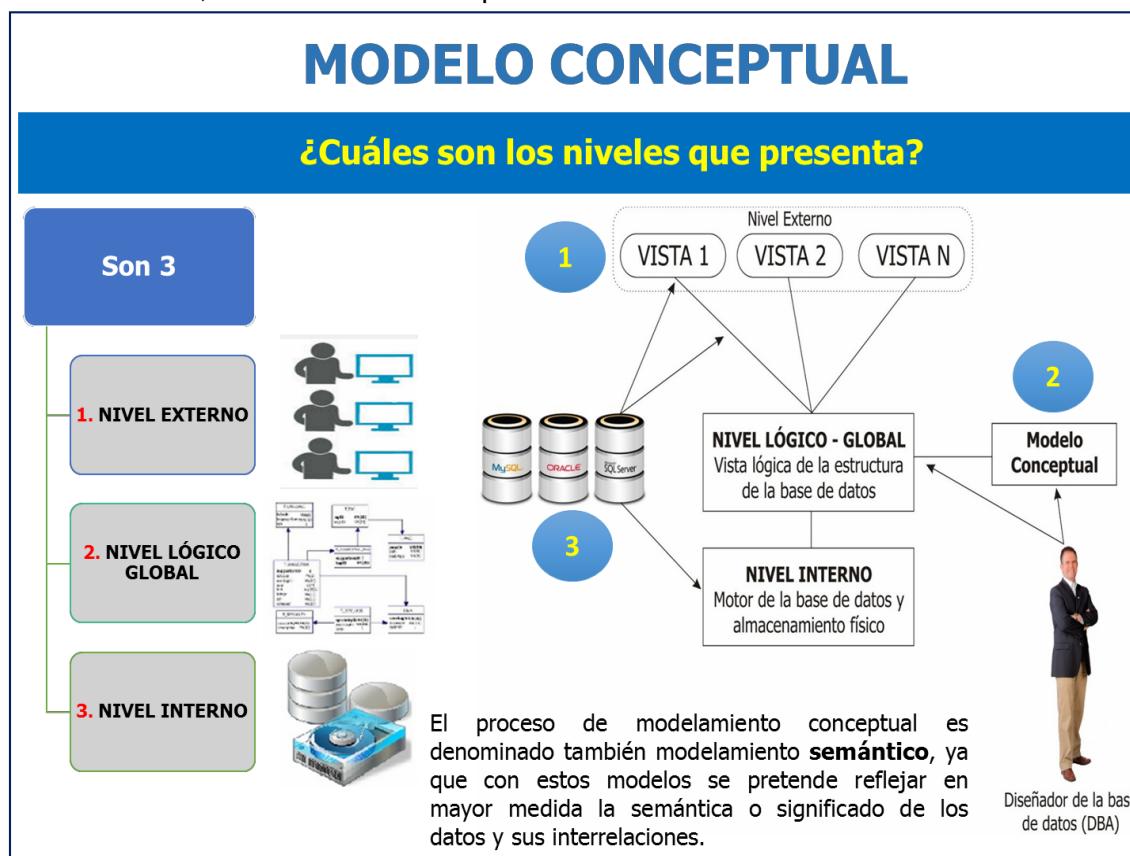
Los **SGBD** existentes utilizan diferentes modelos de datos para la representación en el nivel lógico global. Son comunes a todos ellos las siguientes características:

- La representación de la información se basa en el uso de determinadas estructuras de datos que poseen una capacidad descriptiva limitada; sólo diferencian un rasgo semántico: el tipo de proyección (**1:1, 1: n, n:m**).
- Utilizan una terminología que no es familiar al usuario del sistema, por lo que dificultan la comunicación usuario - diseñador.

Además, cada uno de estos modelos está vinculado con un tipo particular de **SGBD**.

Por todo ello, es necesario tratar con otro tipo de modelo cuando se aborda el problema del diseño de la base de datos, el cual debe superar los problemas anteriores y constituye un nivel de abstracción intermedio entre la realidad informativa y el nivel lógico global de la arquitectura. A este nuevo tipo de modelo se le denomina modelo conceptual.

A continuación, mostraremos los respectivos niveles:



¿Qué es Modelo Entidad Relación?

Este modelo (presentado mediante un diagrama y denominado **DER**) fue propuesto en 1976 y ha encontrado una amplia aceptación como instrumento para modelar el mundo real en el proceso de diseño de las bases de datos. El modelo opera con los conceptos de **entidad** y **relación** que estudiamos anteriormente.

Las **ocurrencias** de **entidades** se clasifican en distintas entidades, tales como "**empleado**", "**departamento**", "**Cliente**", etc. Existirá un predicado asociado con cada entidad que permitirá comparar si una ocurrencia arbitraria pertenece a una entidad dada. Las ocurrencias pueden pertenecer a más de una entidad, o sea, las entidades no son mutuamente disjuntas. Por ejemplo, una ocurrencia de la entidad "**mujeres**" también pertenece a la entidad "**persona**".

Se le llama relación a una asociación matemática entre n entidades.

$$\{ (e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n \}$$

Cada elemento de esa relación es una ocurrencia de relación (e_1, e_2, \dots, e_n), donde las E_i y e_i no tienen que ser necesariamente diferentes. El rol de una entidad en una relación expresa la función que desempeña dicha entidad en la relación.

En la relación "matrimonio" definida entre ocurrencias de la entidad "persona", mejor dicho, "matrimonio" = $\{(e_1, e_2) \mid e_1 \in \text{"persona"}, e_2 \in \text{"persona"}\}$, el primer elemento en el tupla puede aparecer en el rol de "esposo" y el segundo, en el rol de "esposa".

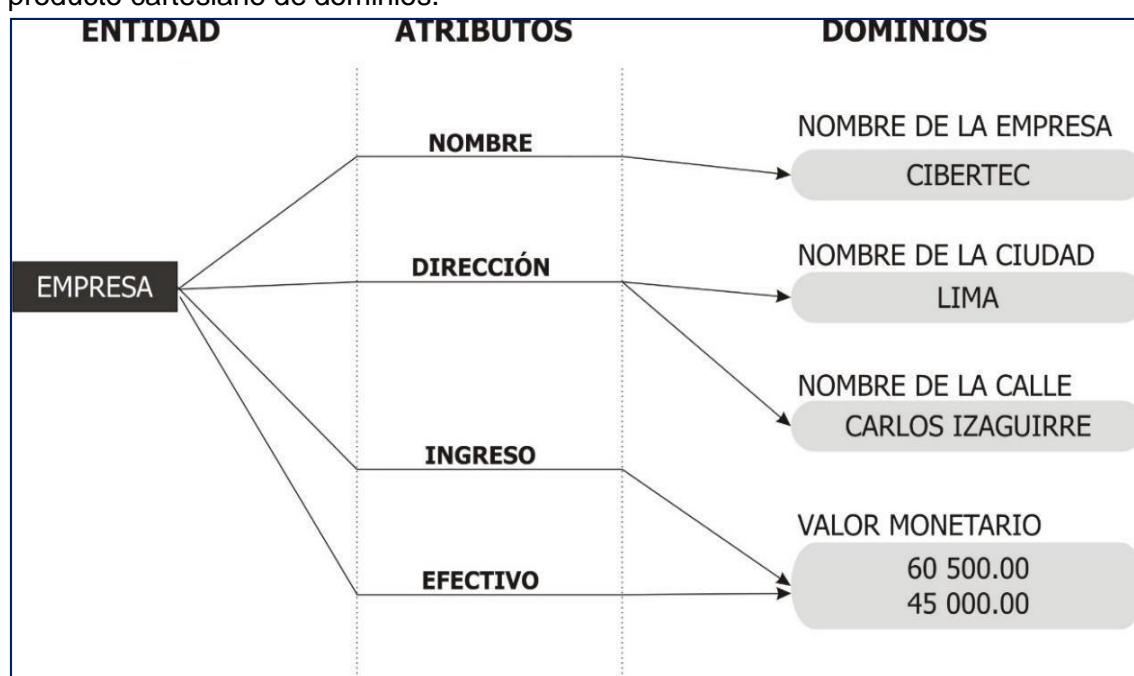
Información adicional sobre una entidad (además de los predicados y las relaciones) se obtiene mediante los atributos asociados con la entidad. Ejemplos de valores que pueden tomar los atributos son: "rojo", "3", "Juan", etc. y ellos se clasifican en dominios mutuamente disjuntos, tales como "color", "edad", "nombre", etc. Un valor de un dominio puede ser equivalente a otro valor en un dominio diferente. Por ejemplo, "100" en el dominio "centímetros" es equivalente a "1" en el dominio "metros".

Un atributo se define en el modelo entidad – relación, como una función matemática que establece una correspondencia desde una entidad o relación hacia un dominio o un producto cartesiano de dominios:

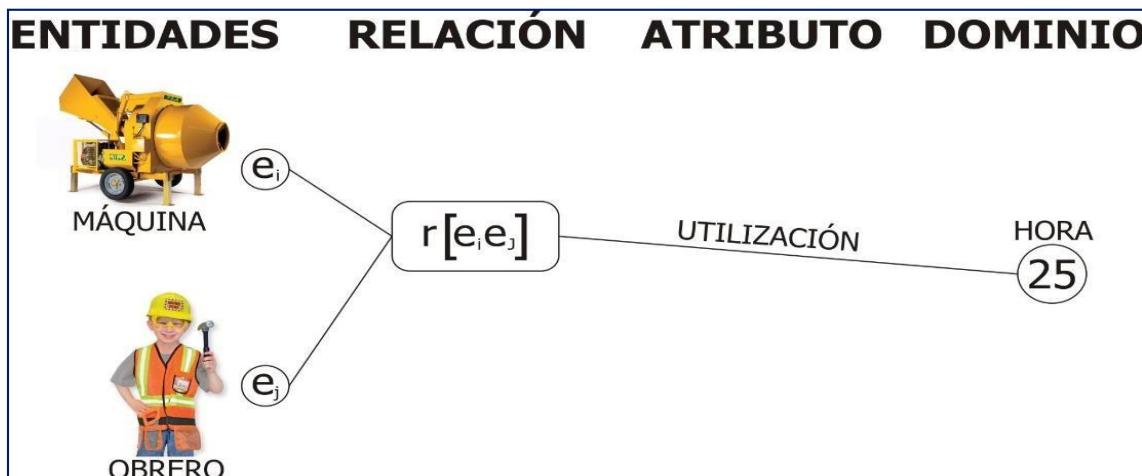
$$\text{atrib}_1: E_i \rightarrow D_{i1} \times D_{i2} \times \dots \times D_{in}$$

$$\text{atrib}_2: R_i \rightarrow D_{i1} \times D_{i2} \times \dots \times D_{in}$$

En la figura siguiente se muestran los atributos definidos para la entidad EMPRESA. El atributo NOMBRE hace corresponder a las ocurrencias de empresa con elementos del dominio NOMBRE DE EMPRESA. El atributo DIRECCIÓN establece una correspondencia desde la entidad EMPRESA hacia el par de dominios NOMBRE DE CIUDAD, NOMBRE DE CALLE. INGRESO Y EFECTIVO establecen ambos una correspondencia desde la entidad EMPRESA hacia el dominio VALOR MONETARIO. Nótese que un atributo se define siempre como una función, por lo que siempre hace corresponder a una ocurrencia dada con un único valor de una tupla, pues se define un producto cartesiano de dominios.



Las relaciones pueden también tener atributos. En la figura siguiente, el atributo **UTILIZACIÓN** define el número de horas que un obrero específico e_j usa una máquina e_i y constituye un atributo de la relación correspondiente. Él no es ni un atributo del **OBRERO** ni de la **MÁQUINA**, ya que su significado depende de la relación entre ellos dos.



Es importante destacar las siguientes características de los atributos en este modelo:

- ✓ Los atributos sólo son correspondencias funcionales. Así, por ejemplo, si tenemos la entidad **AUTOMÓVIL** y el atributo **COLOR**, el hecho de que un auto pueda tener más de un color no se puede representar como un atributo en este modelo.
- ✓ El único hecho que puede ser registrado sobre los valores en este modelo es su pertenencia a un dominio. Si se desea representar otra propiedad, el atributo asociado tiene que ser convertido en una entidad. Por ejemplo, si queremos registrar la longitud de onda de cada color no podemos hacerlo en el modelo entidad - relación, sino convirtiendo el atributo **COLOR** en una entidad.

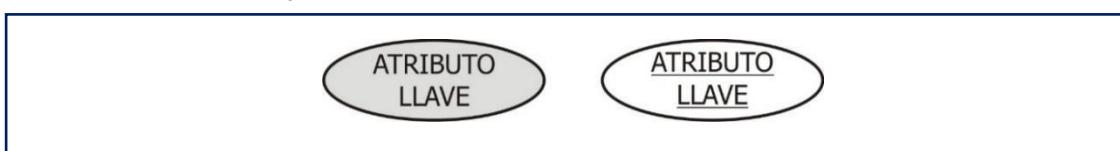
2.2. Tema 6 : Diagrama Entidad - Relación

El modelo entidad-relación tiene asociada una representación gráfica denominada diagrama entidad-relación (**DER**).

En un **DER**, cada entidad se representa mediante un **rectángulo**, cada relación mediante un **rombo** y cada dominio mediante un **ovalo**. Mediante líneas se conectan las entidades con las relaciones, y de la misma manera igual que las entidades con los dominios, representando a los atributos.



“Los **atributos llaves** de las entidades se representan mediante un **ovalo relleno** o con el **nombre subrayado**”.



En ocasiones, una entidad no puede ser identificada únicamente por el valor de sus propios atributos. En estos casos, se utilizan conjuntamente las relaciones con los atributos para lograr la requerida identificación unívoca. Estas entidades reciben el nombre de **entidades débiles** y se representan en el **DER** con un **doble rectángulo**.

ENTIDAD DEBIL

El **modelo entidad-relación** restringe las relaciones a usar para identificar las entidades débiles a relaciones binarias de, a lo sumo, **1: n**. Así, por ejemplo, una ocurrencia de trabajador puede tener n ocurrencias persona-dependiente asociadas, donde, además, la existencia de una ocurrencia en la segunda entidad depende de la existencia de una ocurrencia que le corresponda en la primera entidad.

Por ejemplo, en el modelo que se representa en el **DER** de la figura, habrá personas dependientes de un trabajador sólo si ese trabajador existe. Para indicar esa dependencia en la existencia, se usa una flecha en el **DER**. La llave de una entidad débil se forma combinando la llave de la entidad regular que la determina con algún otro atributo o conjunto de atributos de la débil, que definen únicamente cada entidad débil asociada a una entidad regular dada (una entidad se denomina regular si no es débil).

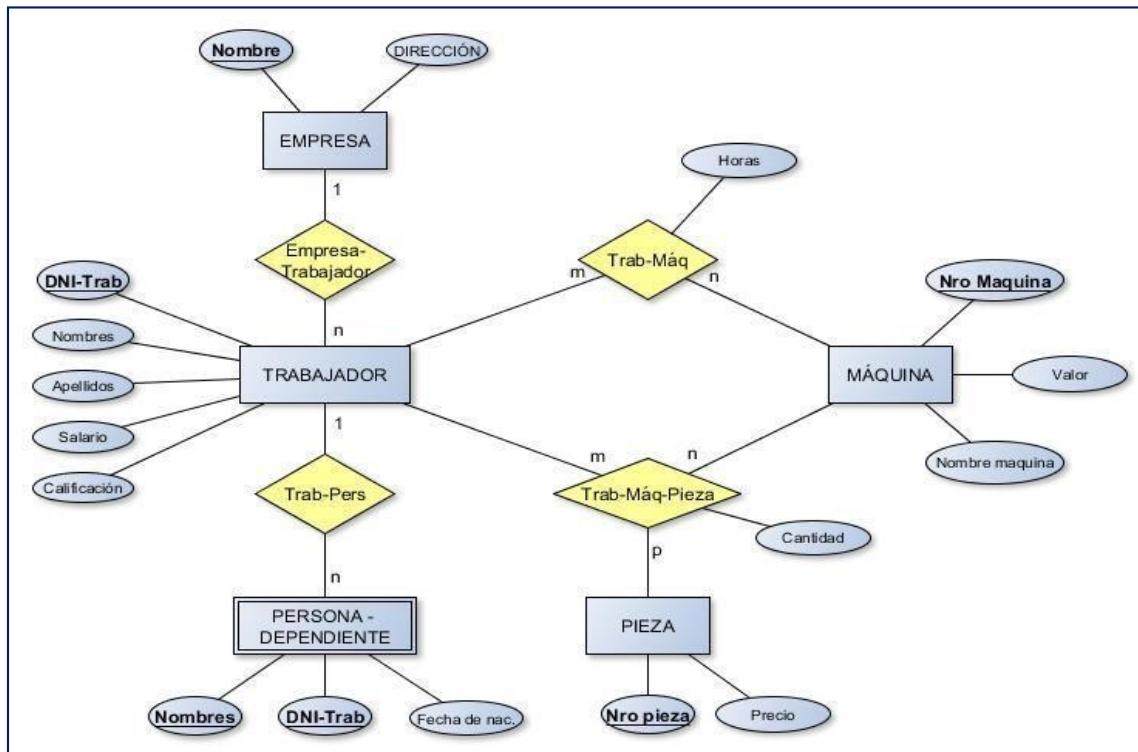


Figura 4: Diagrama Entidad Relación diseñado en yEd 3.16.2.1

Fuente. - Tomado desde la aplicación yEd

Para cada relación se determina su tipo (simple o complejo) y en el **DER** se escribe el tipo de correspondencia, por ejemplo:

- ✓ Una empresa puede tener varios (**n**) trabajadores asociados y un trabajador pertenece a una sola (**1**) empresa.

- ✓ En la relación **trab-máq-pieza**, un trabajador puede trabajar en **n** máquinas, produciendo **p** piezas, una pieza puede ser producida por **m** trabajadores en **n** máquinas y en una máquina pueden trabajar **m** trabajadores produciendo **p** piezas.
- ✓ **m**, **n** y **p** no identifican un número específico, sino solamente el tipo de correspondencia que se establece en la relación (muchos).

Aunque en el **modelo entidad-relación** se define que “**La llave de una relación es la combinación de las llaves de todas las entidades asociadas**”, es conveniente, desde ahora, analizar profundamente esto.

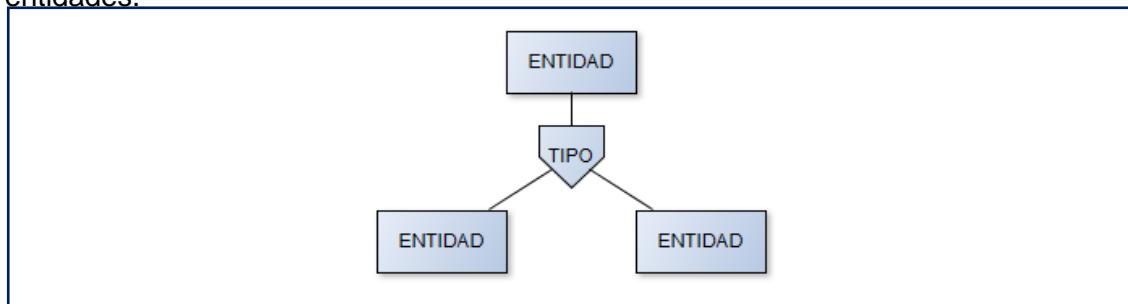
En una relación de **MUCHOS A MUCHOS (n:m)**, efectivamente, “**La llave de la relación está formada por las llaves de las entidades que participan en la relación**”, pues, como a cada ocurrencia de una de las entidades le corresponden varias ocurrencias de la otra entidad y viceversa, es preciso utilizar la identificación de cada una de las entidades que participan en la relación para referirse a una sola ocurrencia de cada una de ellas y, así, referirse a una ocurrencia de la relación. **Por ejemplo**, en la relación **trab-máq** la llave será **DNI-Trab, Nro-Máquina**. Pero en una relación de muchos a uno (**m:1**), la llave de la relación es la llave de la entidad del extremo muchos (**m**), pues a cada ocurrencia de esa entidad le corresponde sólo una ocurrencia de la entidad del otro extremo, por lo que, con la llave de la entidad del extremo muchos están perfectamente determinadas también una única ocurrencia de la entidad del extremo 1. **Por ejemplo**, en la relación **Empresa-trabajador** la llave será **DNI-trab**.

De modo similar, en una relación de **UNO A UNO (1:1)** la llave de la relación está formada por la llave de cualquiera de las dos entidades que participan, pues “**A una ocurrencia de una de ellas le corresponde sólo una ocurrencia de la otra y viceversa**”, por lo que, con la llave de una de las entidades está perfectamente determinada también una única ocurrencia de la otra entidad. Por ejemplo, si en un centro de trabajo, un trabajador que se identifica con DNI-Trab, es jefe de un piso, que se identifica con numpiso, y un piso tiene un jefe, es decir, que la relación es de **1:1**, entonces, la llave de la relación puede ser **DNI-trab o numpiso**.

Una entidad se puede relacionar consigo misma. A estas relaciones se les llama, usualmente, **recursivas o cíclicas**. Es posible extender la capacidad semántica del modelo entidad-relación aplicando sobre sus objetos básicos (entidad y relación) diferentes operaciones, tales como:

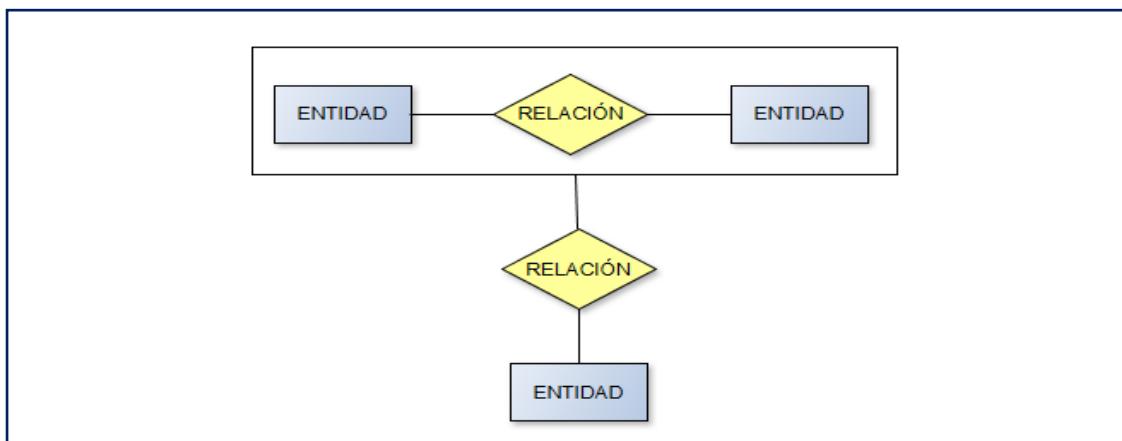
• GENERALIZACIÓN

Permite formar una nueva entidad, mediante la unión de otras entidades. El proceso inverso se denomina especialización y divide una entidad en cierto número de otras entidades.

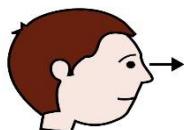


• AGREGACIÓN

Construye una nueva entidad sobre la base de una relación. A las entidades, relaciones y conjuntos definidos hasta ahora les llamaremos tipos básicos para distinguirlos de los nuevos tipos de datos que se obtendrán con las operaciones anteriores.



Veamos cada una de las operaciones:



2.2.1. Entidades: Generalización / Especialización

Si T_1, T_2, \dots, T_n son entidades (que pueden a su vez ser resultado de una generalización), la generalización define una nueva entidad T con el siguiente significado:

$$T = \{ t \mid t \in T_i, 1 \leq i \leq n \}$$

Mejor dicho, para cada ocurrencia t en T existe, al menos, un conjunto T_i que contiene a esa ocurrencia. Por ejemplo, en el **DER** anterior, puede ser necesario distinguir los trabajadores de una empresa de acuerdo a su ocupación como obreros, dirigentes y administrativos. Esto no puede ser representado en el modelo que está representado en el **DER** de la figura anterior y sólo a través del hecho de que la entidad "obrero", por ejemplo, es siempre (mejor dicho, en todo momento) un subconjunto de la entidad "trabajador", podemos deducir cierta clase de dependencia entre los dos tipos.

Usando la generalización podemos obtener un nuevo diagrama como se muestra parcialmente en la figura siguiente:

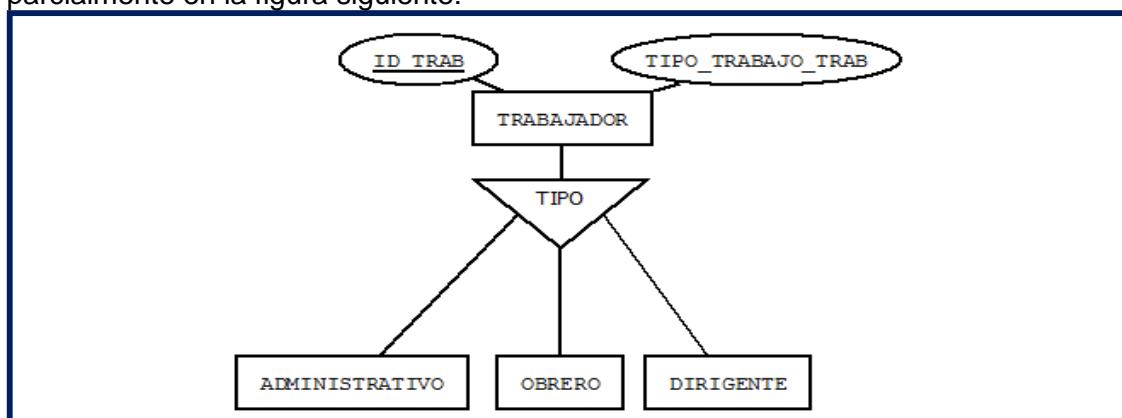


Figura 5: Generalización de trabajador
Fuente. - Tomado desde la aplicación diaW

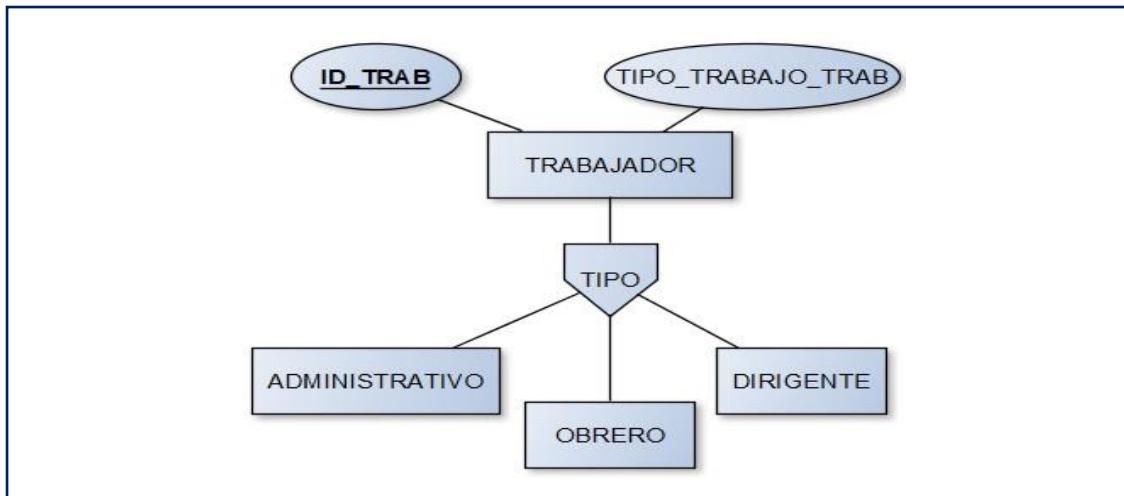


Figura 6: Generalización de la entidad trabajador

Fuente. - Tomado de la aplicación yEd

Nótese que hemos introducido un nuevo atributo para la entidad trabajador. Este atributo nos permite distinguir entre los miembros de diferentes clases de trabajadores.

Si tenemos una entidad **TRABAJADOR** y queremos usar la operación de Especialización como inversa a la generalización, tenemos que especificar "roles" en el modelo, dicho de otra forma, reglas que definen cuándo una ocurrencia de **TRABAJADOR** pertenece a uno u otro componente de la entidad. Entonces, la representación de esta operación en el **DER** se generaliza como se muestra en la siguiente figura:

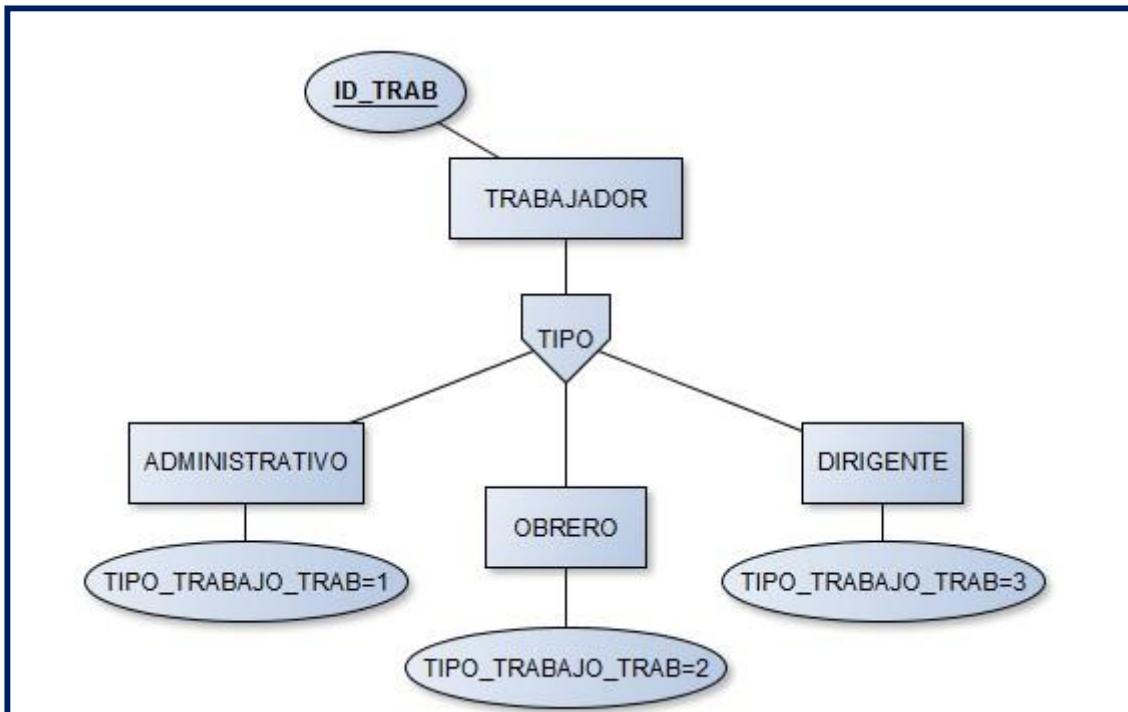


Figura 7: Especialización de la entidad Trabajador

Fuente. - Tomado desde la aplicación yEd

Si para cada ocurrencia de la entidad **Trabajador** nosotros podemos siempre deducir a cuál entidad componente pertenece usando alguna propiedad ya representada, entonces, no es necesario introducir un nuevo atributo **Tipo de Trabajo**.

Las reglas que definen la especialización de una entidad se denominan "caracterizaciones". Por ejemplo, **Tipo de Trabajo = 1** es la caracterización de la entidad Administrativo dentro de la entidad Trabajador.

En una Generalización / Especialización, los atributos y relaciones de la entidad "generalizada" son heredados por las entidades componentes (entidades especializadas). La llave de una especialización es la llave de la generalización. Además, se pueden definir nuevos atributos y relaciones para cada entidad especializada. Por ejemplo, la relación Obrero-Máquina se define ahora sólo para la entidad especializada Obrero, componente de la entidad generalizada Trabajador:

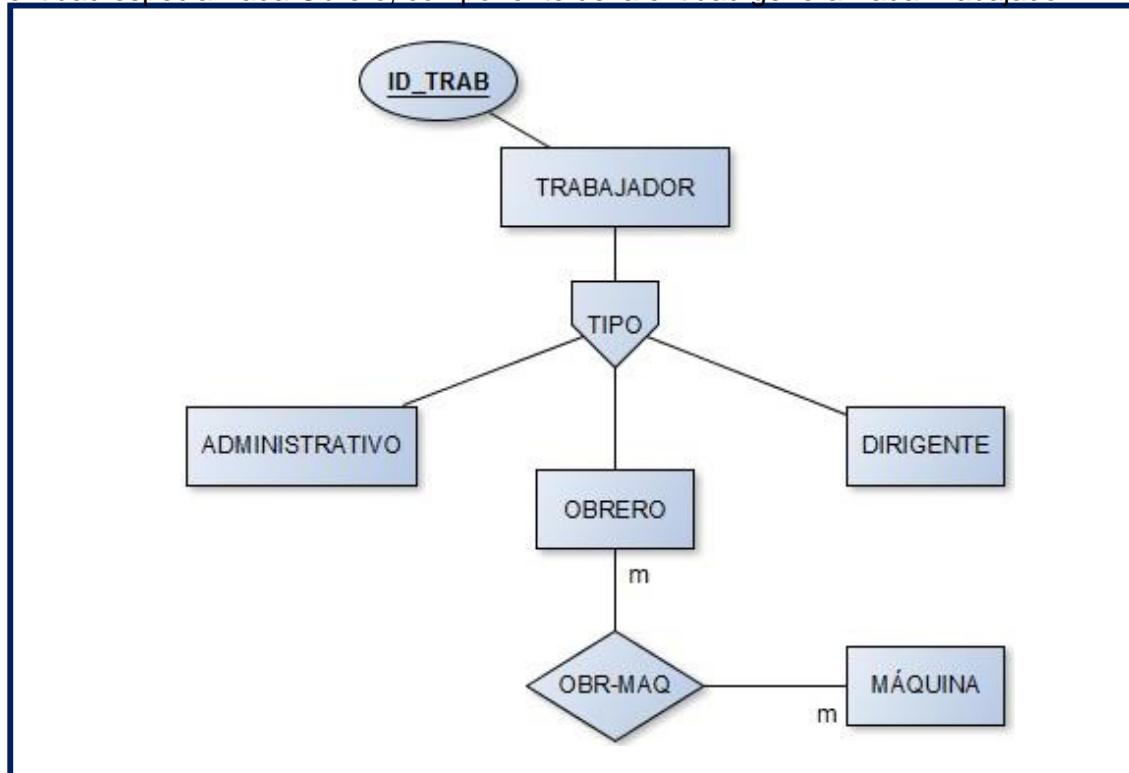


Figura 8: Generalización de la entidad Trabajador
Fuente. - Tomado desde la aplicación yEd

Si bien es cierto que, según lo visto anteriormente, las operaciones de Generalización y Especialización pueden denotarse de modo diferente, no es menos cierto que, con la notación empleada para la generalización, pueden expresarse las entidades generalizadas y especializadas perfectamente y es ésta la empleada normalmente.

Es importante agregar algo más a lo visto hasta ahora para poder expresar las siguientes situaciones que se presentan:

- Las ocurrencias de las especializaciones pueden abarcar o no el universo de las ocurrencias de la generalización; es decir, la totalidad de las ocurrencias de la generalización puede o no estar contenidas en alguna o algunas de las especializaciones. Por lo tanto, las especializaciones pueden ser totales (T) o parciales (P).
- Una ocurrencia de la generalizada puede o no estar en más de un conjunto T_i , o lo que es lo mismo, la intersección entre algunos de los conjuntos T_i puede o no ser vacía. Es decir, las especializaciones pueden ser solapadas (S) o disjuntas (D).

Es por ello que, en el **DER**, se añade en cada generalización, entre paréntesis, la especificación:

- (T, S): indicando que la especialización realizada es total y solapada
- (T, D): indicando que la especialización realizada es total y disjunta
- (P, S): indicando que la especialización realizada es parcial y solapada
- (P, D): indicando que la especialización realizada es parcial y disjunta

Entonces, el ejemplo visto anteriormente quedaría de la siguiente manera:

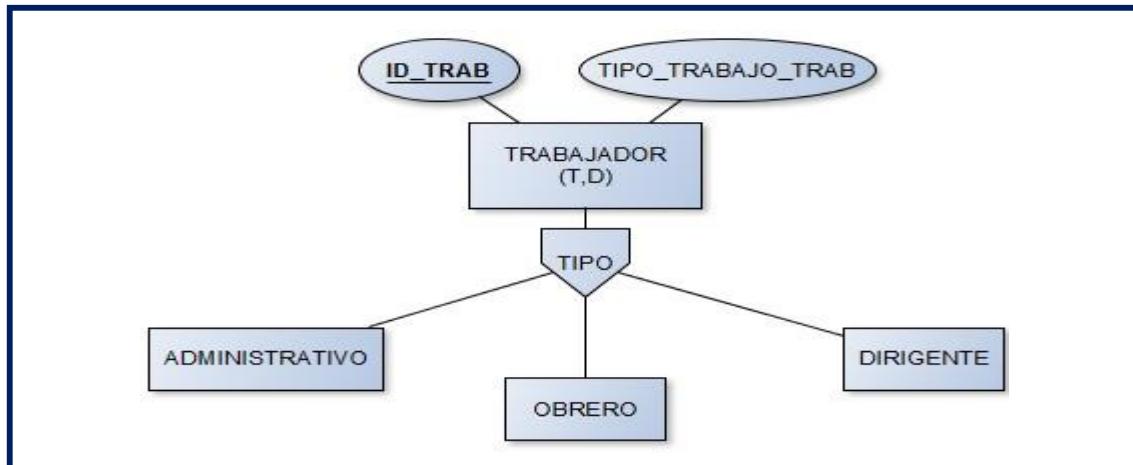


Figura 9: Generalización de la entidad Trabajador

Fuente. - Tomado desde la aplicación yEd

Donde:

- **T(total)**: ya que todo trabajador en el ejemplo es administrativo o dirigente u obrero.
- **D(disjunto)**: Un trabajador pertenece solo a una de las especializaciones.

Otro ejemplo de Generalización/Especialización podría ser el caso de **ESTUDIANTE**, **PRACTICANTE** y **BECADO**. Un **PRACTICANTE** es un caso especial de **ESTUDIANTE**. Lo mismo ocurre con **BECADO**. Pero un **PRACTICANTE** también puede ser **BECADO**. Hay muchos Estudiantes que no son **PRACTICANTES** ni **BECADOS**. Obviando los atributos en el **DER**, esta situación se representaría del modo siguiente:

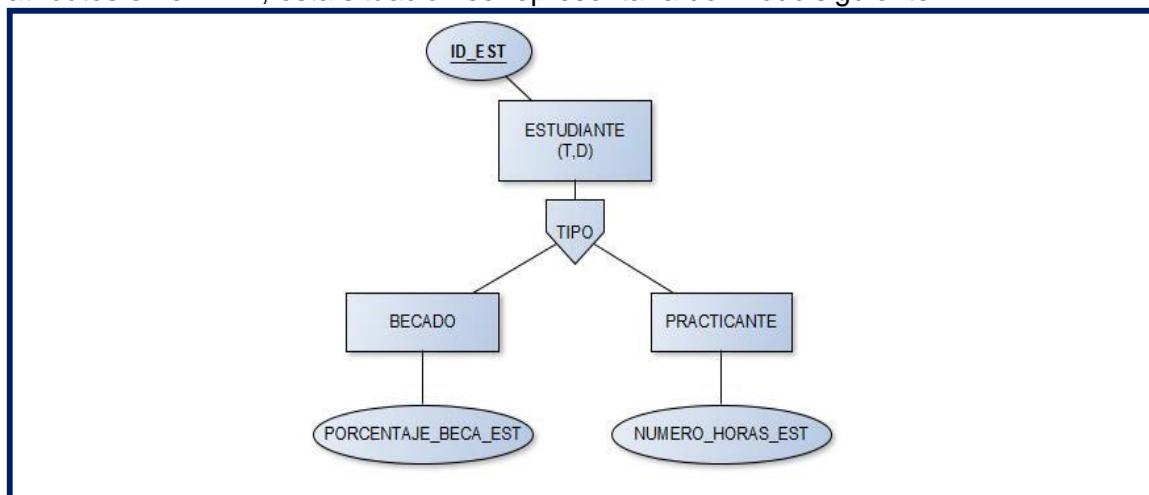


Figura 10: Generalización de la entidad Estudiante

Fuente. - Tomado desde la aplicación yEd

EJEMPLO:

Tomando el ejemplo del libro de procesamiento de base de datos de David M. Kroenke.

Donde:

Se tiene las entidades Cta_Ahorro y Cta_Cheques, ambas tienen los atributos semejantes de No_Cta y Saldo, aunque además de estos dos atributos, Cta_Ahorro tiene el atributo Tasa_Interes y Cta_Cheques el atributo Saldo_Deudor. De todos estos atributos podemos juntar (generalizar) No_Cta y Saldo que son iguales en ambas entidades.

Entonces tenemos:

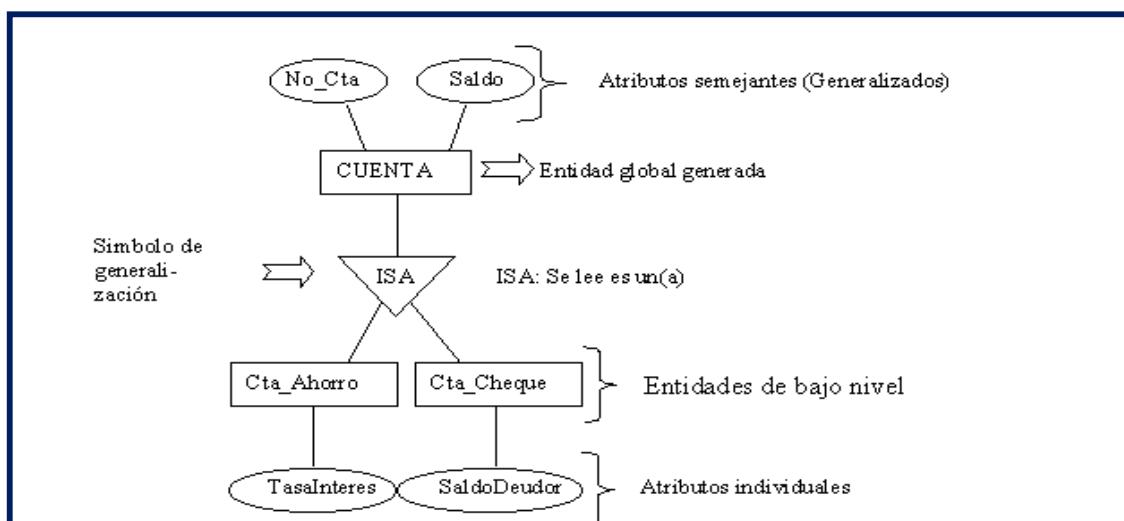


Figura 11: Generalización de la entidad Cuenta

Fuente. - Tomado desde https://disjuntos.wikispaces.com/file/view/fig2_10.gif/265127352/fig2_10.gif

Podemos leer esta gráfica como: La entidad Cta_Ahorro hereda de la entidad CUENTA los atributos No_Cta y saldo, además del atributo de TasaInteres, de forma semejante Cta_Cheque tiene los atributos de No_Cta, Saldo y SaldoDeudor.

Como podemos observar la Generalización trata de eliminar la redundancia (repetición) de atributos, al englobar los atributos semejantes. La entidad(es) de bajo nivel cuentan (heredan) todos los atributos correspondientes.

2.2.2. Entidad Fuerte/Débil

- **Entidad Fuerte**

Lo constituyen **las tablas principales** de la base de datos que contienen los registros principales del sistema de información y que requieren de entidades o tablas auxiliares para completar su descripción o información. Por ejemplo, la tabla **usuario** es una entidad fuerte en relación a la tabla **tipos de usuarios**, que es una entidad débil dada su condición auxiliar para clasificar a los usuarios registrados en la biblioteca.

- **Entidad Débil**

Son entidades débiles a **las tablas auxiliares** de una tabla principal a la que completan o complementan con la información de sus registros relacionados. Por ejemplo, también son consideradas entidades débiles las tablas intermedias que sirven para compartir información de varias tablas principales.

Cuando una entidad participa en una relación puede adquirir un papel fuerte o débil. Una entidad débil es aquella que no puede existir sin participar en la relación; es decir, aquella que no puede ser únicamente identificada solamente por sus atributos. Una entidad fuerte (también conocida como entidad regular) es aquella que sí puede ser identificada únicamente. En los casos en que se requiera, se puede dar que una entidad fuerte "preste" algunos de sus atributos a una entidad débil para que esta última se pueda identificar.

Las entidades débiles se representan- mediante un **doble rectángulo**; es decir, un rectángulo con doble línea. Se puede hablar de la existencia de 2 tipos de dependencias en las entidades débiles:

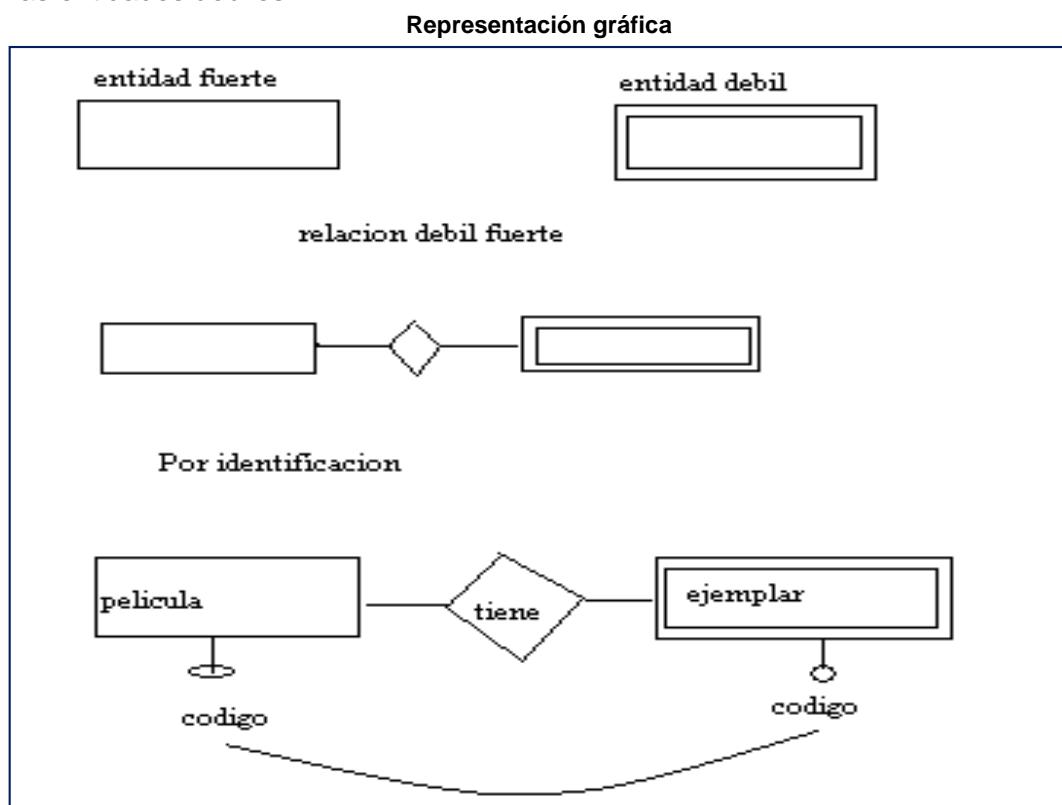


Figura 12: entidad fuerte / débil

Fuente. - Tomado desde http://2.bp.blogspot.com/_i6sp-fiAP30/TIAemWn2cmI/AAAAAAAEEI/rqjHCztXkY8/s400/entidadades+fuertes+y+debiles.PNG

2.2.3. Ejercicios de aplicación

Ejercicio 1

A manera de **ejemplo**, se muestra a continuación un **DER** que representa al alumno de Medicina como entidad débil de grupo, ya que el alumno se identifica por un número consecutivo dentro del grupo al que pertenece y que, por lo tanto, puede repetirse en distintos grupos, por lo que, para identificar a un alumno, es preciso decir "**el alumno número n del grupo código g**". A la vez, la entidad alumnomedicina es una generalización que tiene como especializaciones a los alumnos de 3er. ciclo o menos y a los alumnos de 4to. ciclo o más.

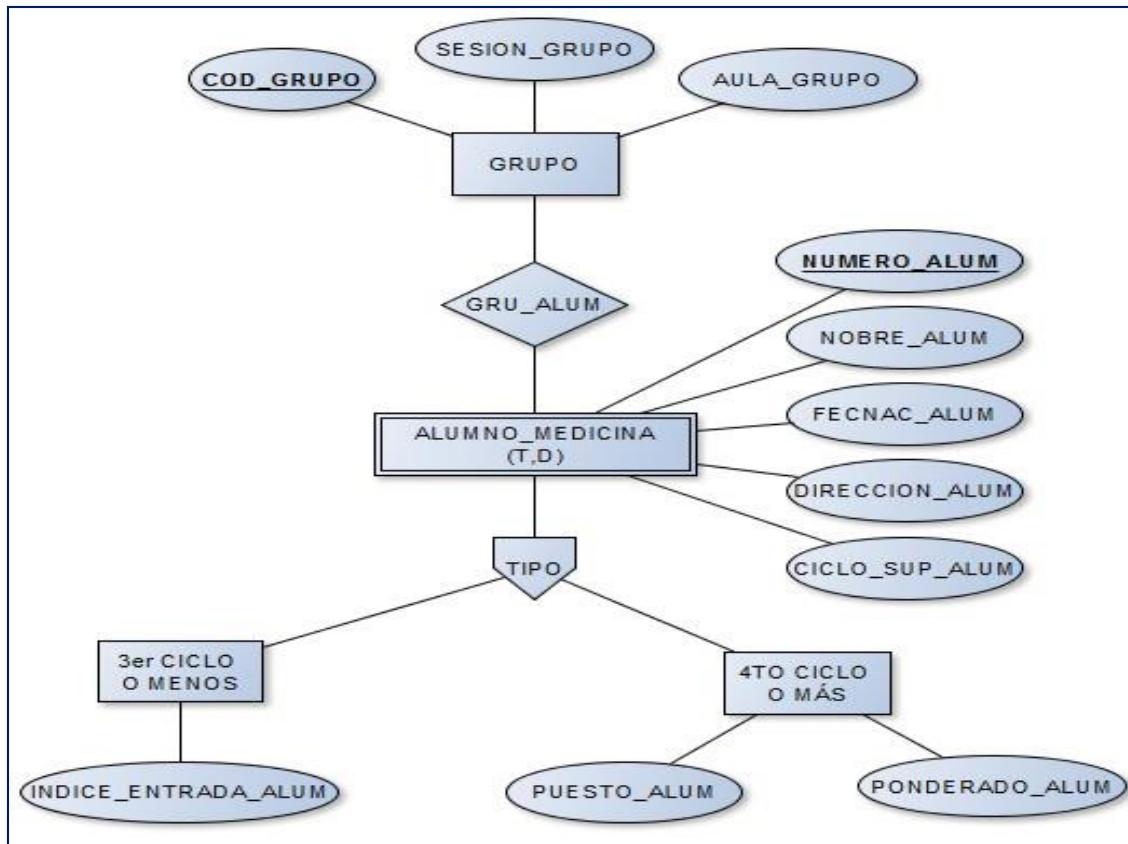


Figura 13: Representación de la entidad débil
Fuente. - Tomado desde la aplicación yEd

Ejercicio 2

Por ejemplo, en un video-club lo que realmente se alquila a los clientes no son películas, sino las copias de películas que tiene el video-club adquiridas.

En el modelo Entidad-Relación se representa de la siguiente manera



Figura 14: Representación de la entidad débil
Fuente. - Tomado desde la aplicación yEd

2.2.4. Relaciones recursivas

Relación donde la entidad se relaciona consigo mismo. Las multiplicidades se colocan respecto a los nombres de rol.



Figura 15: Representación de la recursividad supervisor
Fuente. - Tomado desde la aplicación yEd

2.2.5. Agregaciones

Obsérvese en el ejemplo que representa la situación de la producción en las empresas, que la relación **Trab-Máq-Pieza** representa la idea de que una actividad en la empresa se describe en términos de "**un obrero en alguna máquina produce una pieza dada en alguna cantidad específica**".

Sin embargo, la misma situación puede ser vista de forma algo diferente. En la empresa, las máquinas pueden estar asignadas a los obreros y estos "**equipos**" producir piezas en cierta cantidad. En el modelo entidad-relación original esta situación no hubiera podido ser modelada correctamente, ya que una relación no puede relacionarse con otra relación o entidad.

Con la operación de Agregación esta situación se resuelve fácilmente, tal y como se muestra en la figura siguiente:

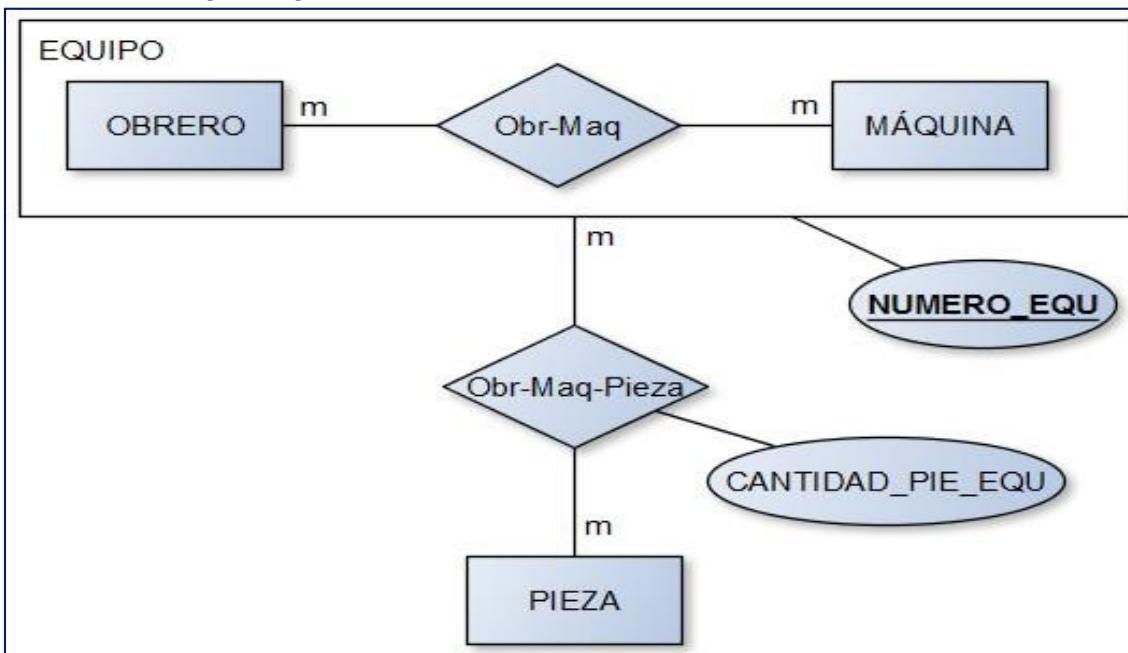


Figura 16: Implementación de la Agregación Equipo
Fuente. - Tomado desde la aplicación yEd

La agregación se define de la siguiente forma:

Si T_1, T_2, \dots, T_n son entidades, la operación define una nueva entidad T con el significado siguiente:

$$T = \{t \mid \exists t_1, t_2, \dots, t_n (t_1 \in T_1 \wedge t_2 \in T_2 \dots \wedge t_n \in T_n \wedge (t_1, t_2, \dots, t_n) = t)\}$$

Mejor dicho, las nuevas ocurrencias se forman como tuplas de ocurrencias de las entidades componentes. Para que la operación tenga sentido, las entidades T_1, T_2, \dots, T_n tienen que formar parte en alguna relación común y esa relación siempre será incluida en la representación de la entidad generada (entidad agregada).

A la nueva entidad se le pueden asignar atributos. También puede tomar parte en cualquier relación. **Otro ejemplo** de Agregación se muestra a continuación:

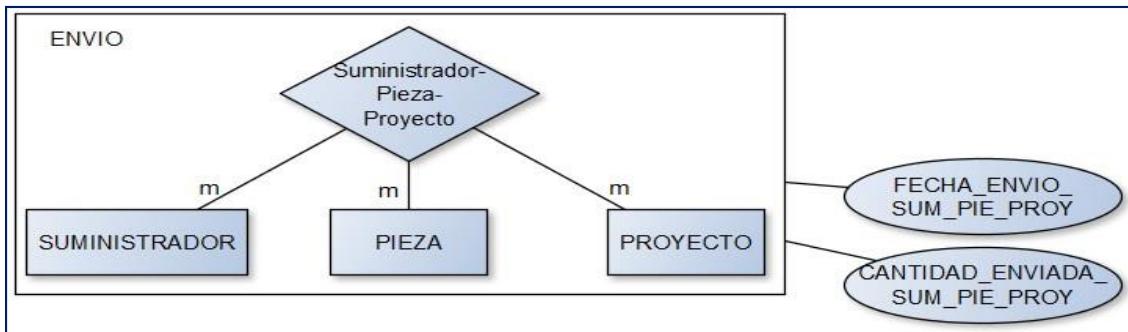


Figura 17: Implementación de la agregación Envío
Fuente. - Tomado desde la aplicación yEd

La nueva entidad **ENVÍO** se define como una agregación de tres entidades: **Suministrador**, **Pieza** y **Proyecto** con los nuevos atributos Fecha del Envío y Cantidad Enviada.

Hay una diferencia importante entre estos dos atributos: está claro que la **Fecha del Envío** no puede pertenecer a ninguna de las entidades componentes; sin embargo, la Cantidad Enviada se refiere claramente a las piezas. Diremos entonces, que la Cantidad Enviada es una "caracterización" de la entidad **PIEZA** con respecto al **ENVÍO**.

La llave de una entidad agregada es la llave de la relación que la origina; excepto en el caso en que se defina un identificador (llave) para ella. En el ejemplo anterior del **ENVÍO**, la llave de la entidad agregada **ENVÍO** está formada por la llave del suministrador, más la llave de la pieza, más la llave del proyecto. Pero la situación del ejemplo podría tener la variante que se muestra en la siguiente figura, donde se ha definido un número de envío (**NUM_ENV**) que identifica al envío y que, por lo tanto, pasa a ser la llave de la entidad agregada:

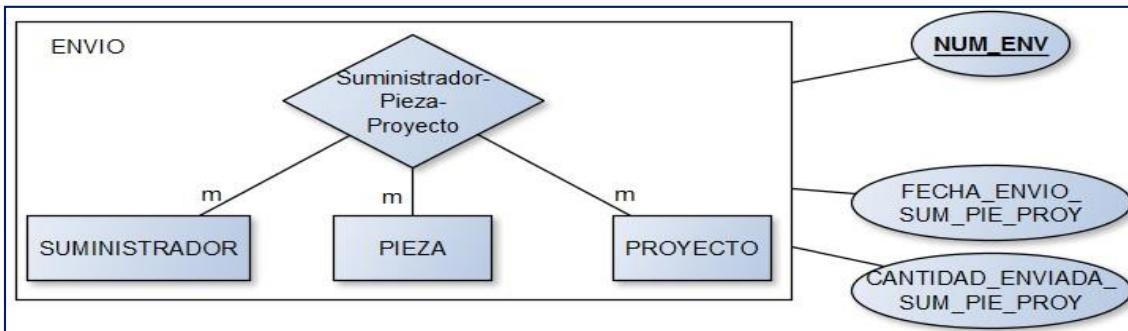


Figura 18: Implementación de la agregación Envío con sus atributos
Fuente. - Tomado desde la aplicación yEd

Para el modelo entidad - relación, incluyendo las dos operaciones estudiadas, pueden plantearse una serie de restricciones de integridad:

- Al aplicar la generalización/especialización, una entidad puede pertenecer a una jerarquía de diferentes entidades. Por ejemplo, las entidades **PERSONA**, **TRABAJADOR**, **OBRERO** forman una jerarquía de entidades, sucesivamente más especializadas. Entonces, una entidad existente en un nivel dado, tiene que existir en todos los niveles superiores. De forma inversa, si una entidad se elimina de un conjunto en un nivel dado, debe ser eliminada también en los niveles más bajos.
- La agregación constituye una entidad agregada sobre la base de una relación, por lo que dicha entidad se comportará de forma similar a como se comporta la relación. Entonces, para que una ocurrencia de la agregación exista, deben existir las

ocurrencias de todas las entidades que toman parte en la relación. Lo inverso no tiene que ocurrir necesariamente, ya que, por ejemplo, en el caso visto del **ENVÍO** pueden existir suministradores que no abastecan a ningún proyecto, sino que se registran como tales, porque en determinado momento pudieran estar activos. Desde luego, si la política de la organización es tal que un suministrador se considera como tal solo si realmente suministra piezas a algún proyecto, entonces, la existencia de, al menos, una ocurrencia de la entidad agregada **ENVÍO** para un suministrador es indispensable para la existencia de la ocurrencia de ese suministrador en la entidad **SUMINISTRADOR**.



Finalmente, es importante señalar que en un **DER** pueden combinarse todos los elementos y operaciones explicadas anteriormente para representar adecuadamente un caso de la realidad, de modo que, por ejemplo, en una agregación puede participar una entidad débil o una generalizada; una especialización puede ser, a su vez, la generalización de otras especializaciones; una entidad débil puede ser, a su vez, una generalizada o ser débil de una generalizada, etc.

2.2.6. Ejercicios integradores

CASO DESARROLLADO: LIGA DE SURCO

La Liga de Surco requiere controlar la constitución de los diferentes equipos deportivos del distrito y de esta manera programar torneos que les permitan mejorar su calidad deportiva. Para ello, ha decidido crear una base de datos.

La liga cuenta con diferentes **clubes** de los cuales se tiene su nombre, fecha de creación, dirección y número de locales. Los clubes tienen distintos tipos de **jugadores** contratados. De los jugadores se conoce su código, el cual se puede repetir para diferentes clubes, los nombres y apellidos, dirección, sexo y fecha de nacimiento, entre otros datos. Cabe mencionar que un jugador es **capitán** de otros jugadores. Ello implicará que deba ser capacitado en cursos de liderazgo y coaching deportivo.

Asimismo, la liga tiene **empleados** de dos tipos: **administrativos** y **técnicos**. De los empleados se almacena un código, los nombres y apellidos, dirección, sexo, fecha de nacimiento y teléfono fijo y celular. Es importante mencionar que para los empleados de tipo Administrativos se almacena su nivel (pregrado o postgrado) y en el caso de los Técnicos, la especialidad deportiva (fútbol, voleibol, natación, etc.) La liga asigna un **Técnico** un grupo de jugadores y estos pueden tener diferentes Técnicos durante la etapa de jugadores, lo cual constituye un **Equipo**; de este se almacena la categoría (de acuerdo a la fecha de nacimiento del jugador, como Sub-15, etc.) y la disciplina.

Los empleados administrativos elaboran varios contratos de los cuales se guarda el número, la fecha de inicio y fin, entre otros datos. Los contratos son confeccionados para los técnicos.

Finalmente, la liga programa a los equipos en diferentes torneos para que eleven su nivel deportivo controlando la cantidad de participaciones que tiene un determinado equipo. Del torneo se registra el nombre del torneo, las fechas de inicio y fin, así como la disciplina correspondiente.

Realice las siguientes actividades:

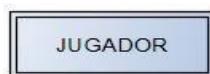
- Identifique las entidades fuertes
- Identifique las entidades débiles
- Identifique las relaciones entre las entidades
- Liste los atributos por cada entidad
- Identifique las llaves simples y complejas de todas las entidades
- Determine la generalización y agregación del caso
- Diseñe el DER final usando yEd.

Solución:

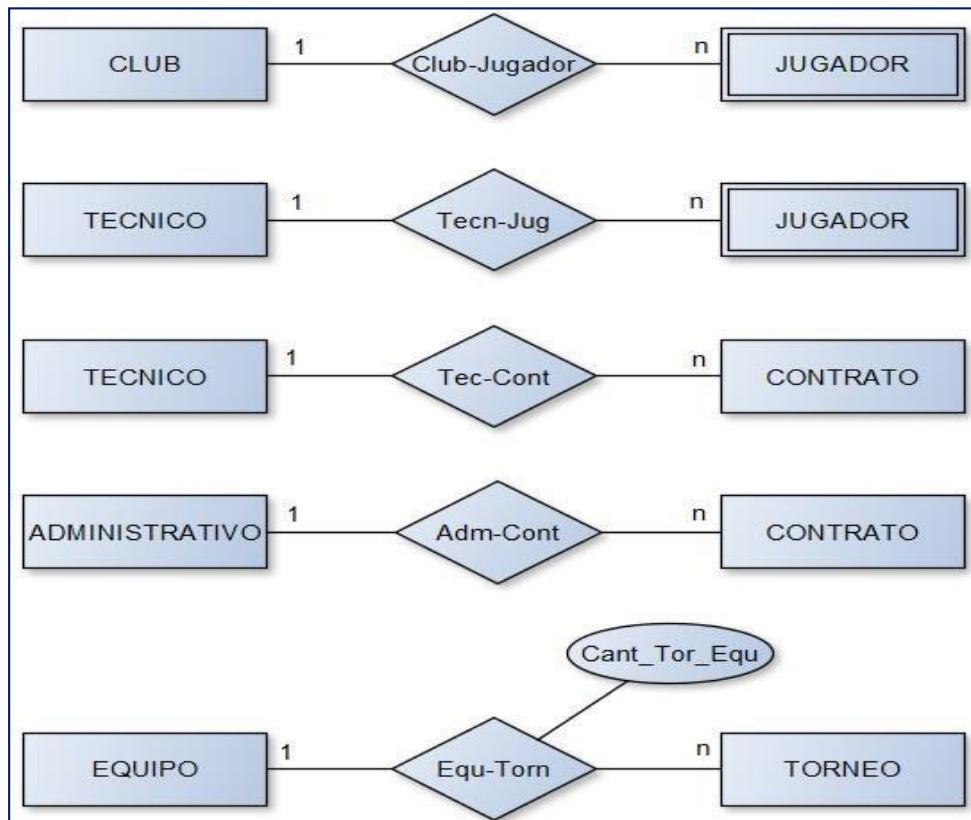
a. Determinando las entidades fuertes:

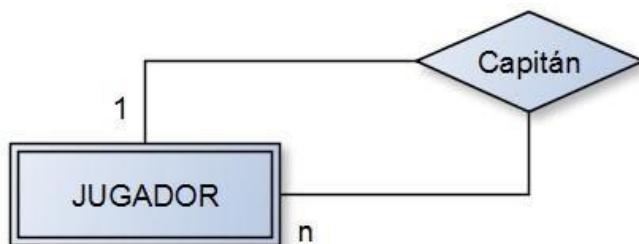


b. Determinando las entidades débiles



c. Identificando las relaciones





d. Identificando los atributos

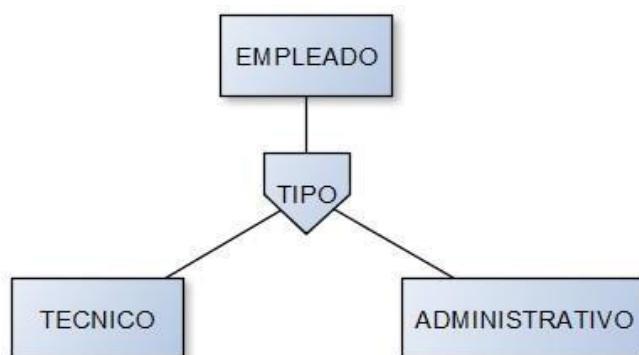
ENTIDAD	ATRIBUTOS
CLUB	cod_club, nom_club, fec_cre_club, dir_club, nro_local_club
JUGADOR	cod_jug, nom_jug, ape_pat_jug, ape_mat_jug, dir_jug, sex_jug, fecnac_jug
EMPLEADO	cod_emp, nom_emp, ape_pat_emp, ape_mat_emp, dir_emp, sex_emp, fecnac_emp, tel_fijo_emp, tel_cel_emp
TECNICO	esp_tec_emp
ADMINISTRATIVO	niv_adm_emp
EQUIPO	cat_equ, disc_equ
CONTRATO	num_cont, fec_ini_con, fec_ter_con
TORNEO	cod_tor, nom_tor, fec_ini_tor, fec_ter_tor, disc_tor

e. Determinando los identificadores o claves (simples o complejas) de cada entidad:

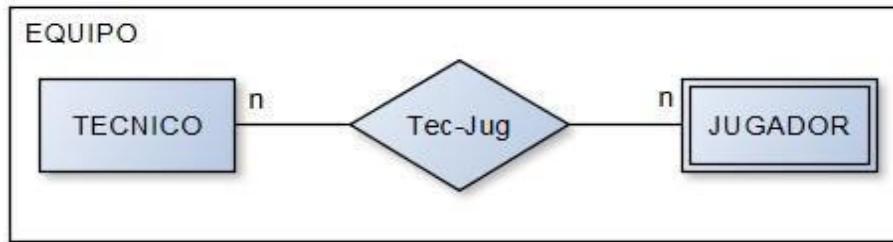
ENTIDAD	ATRIBUTO IDENTIFICADOR
CLUB	cod_club
JUGADOR	cod_club, cod_jug
EMPLEADO	cod_emp
EQUIPO	cod_club, cod_jug, cod_emp
CONTRATO	num_cont
TORNEO	cod_tor

f. Determinando la generalización y agregación

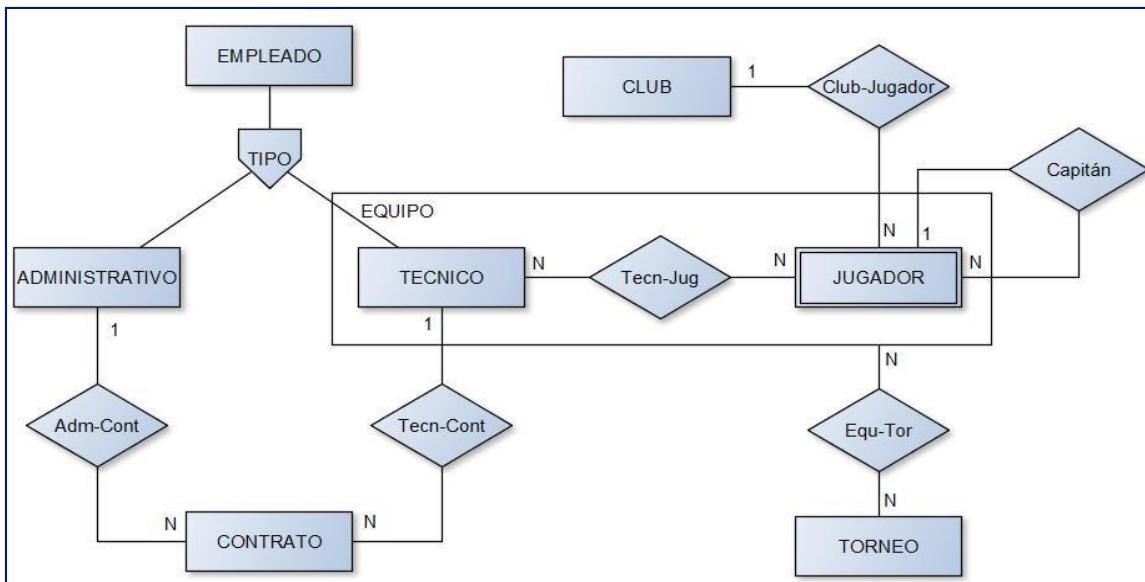
Generalización



Agregación



g. Diagrama de entidad relación (DER)



NOTA: En el DER se deberá agregar los atributos e identificadores de cada entidad, así como en la relación de EQUIPO – TORNEO (E - T) el atributo cantidad de torneos en que participa un equipo (CANT_TOR_EQU), como se vio anteriormente.

El siguiente caso debe desarrollar el docente con los estudiantes.

CASO PROPUESTO 01: Biblioteca Nacional

La Biblioteca Nacional del Perú desea efectuar el control de préstamos de los libros y cubículos a sus diferentes usuarios, para lo cual lo ha contratado a fin de que diseñe y cree una base de datos.

Con respecto a los libros y cubículos son considerados “Recursos” de la biblioteca de los cuales se tiene un código que los identifica. Los libros son prestados a través de una forma denominada Solicitud de Préstamo, de tal manera que un tipo de libro puede ser pedido en varias solicitudes de préstamo, dado que se controla la cantidad de libros existentes (stock). De los libros se almacena su nombre, edición, fecha de edición, cantidad, autor y un breve resumen del libro y de los cubículos se registra su capacidad y cantidad de equipos con que cuenta. Para llevar el control de los libros prestados se registra la fecha de inicio y fin del préstamo, así como el correspondiente control de la devolución.

Así mismo, la biblioteca cuenta con dos (2) tipos de empleados: contratados y practicantes. De los empleados se guarda el código, nombre, apellidos, sexo,

dirección, fecha de nacimiento, documento de identidad y el tipo del trabajador. Una solicitud de préstamo es atendida únicamente por un empleado del tipo contratado, dado que debe dar su autorización mediante la firma de dicho documento. De la solicitud de préstamo se almacena el número de la misma, la fecha de solicitud y datos de los usuarios, libros y empleados.

Sin embargo, los pedidos de los cubículos se efectúan a través de la Internet generándose un número único para su identificación, siendo tramitado exclusivamente por empleados del tipo practicantes (por no requerir una firma física) para aquellos usuarios que lo requieran. De los pedidos de cubículos se registra la fecha del préstamo, el turno solicitado y su correspondiente aprobación. Es importante mencionar que de los empleados contratados se almacena la fecha de inicio del contrato y de los practicantes las fechas de inicio y fin de las prácticas.

Asimismo, los usuarios pueden ser de dos tipos: alumnos y profesores, los cuales son de diferentes institutos o colegios, y pueden generar varias solicitudes de préstamo para diferentes libros o pedidos de cubículos. De los usuarios se almacena su código, nombre, apellidos, dirección, teléfono de casa y documento de identidad.

Finalmente, la biblioteca aplica sanciones basadas en el tiempo que excedió la entrega de uno o varios libros. Las sanciones son de tres tipos:

- Definitiva, por haber perdido uno o más libros.
- Parcial, cuya duración es de un mes de suspensión, por haber excedido la fecha máxima del préstamo en una semana.
- Inicial, cuya duración es de una semana, por haber excedido la fecha máxima del préstamo.

De las sanciones se guarda el tipo de la sanción, fecha inicio, fecha término.

Realice las siguientes actividades:

Identifique las entidades fuertes

- a) Identifique las entidades débiles
- b) Identifique las relaciones entre las entidades
- c) Liste los atributos por cada entidad
- d) Identifique las llaves simples y complejas de todas las entidades
- e) Determine la generalización y agregación del caso
- f) Diseñe el DER final usando yEd.

“La educación es el arma más poderosa que puedes usar para cambiar el mundo”

Nelson Mandela

Autoaprendizaje

Nota: *El siguiente caso deben desarrollar solo los estudiantes, para comprobar su aprendizaje esperado.*

Caso Propuesto 01: Movimiento “Mercantil”

Se desea diseñar una base de datos sobre el movimiento mercantil de un organismo en un año. En el organismo existen mercancías de las que se conoce su código, nombre y unidad de medida. Las mercancías proceden de diferentes países de los que se sabe nombre y tipo de moneda. Para la transportación de las mercancías existen diversas formas, cada una de las cuales se caracteriza por su tipo (barco, avión, tren, etc.) y tarifa.

Así mismo, para cada mercancía de diferentes países existen diferentes formas de transportación; para cada país existen diferentes mercancías que son transportadas en diferentes formas de transportación; y una forma de transportación puede serlo de diferentes mercancías de diferentes países. Una mercancía procedente de un país transportada de una forma dada constituye un embarque y para éste se conoce su fecha de arribo y cantidad.

Por otra parte, un embarque se distribuye entre diferentes almacenes y en un almacén se tienen diferentes embarques, cada uno en cierta cantidad. De cada almacén se tiene su código y dirección. Un almacén distribuye los productos entre diferentes empresas y cada empresa recibe productos de diferentes almacenes. Una empresa se caracteriza por su número, nombre y rama económica; a su vez, las empresas establecen relaciones contractuales entre sí. Entre dos empresas dadas sólo se puede establecer un contrato anual. De cada contrato se conoce su número, valor y fecha de vencimiento.

Realice las siguientes actividades:

- a) Identifique las entidades fuertes
- b) Identifique las entidades débiles
- c) Identifique las relaciones entre las entidades
- d) Liste los atributos por cada entidad
- e) Identifique las llaves simples y complejas de todas las entidades
- f) Determine la generalización y agregación del caso
- g) Diseñe el DER final usando yEd.

Caso Propuesto 02: Industria Mecánica

En una empresa de la industria mecánica se fabrican distintos tipos de piezas. Para cada una se conoce su código, que la identifica, descripción y peso unitario. Sobre cada pieza se realizan distintas operaciones (corte, fresado, etc.) acerca de las cuales se conoce su código y nombre. En su proceso de fabricación, una pieza pasa por diferentes equipos que pueden realizar diferentes operaciones, una operación puede ser realizada en diferentes equipos a diferentes piezas y en un equipo se realizan diferentes operaciones sobre diferentes piezas. De cada equipo se conoce su nombre, modelo y país de procedencia. Una operación realizada sobre una pieza en un equipo dado constituye una norma que se identifica por un código, y tiene asociados una descripción y un tiempo de ejecución.

En la empresa trabajan obreros de dos tipos distintos: directos y auxiliares. Un obrero directo está vinculado con el cumplimiento de diferentes normas y una norma puede ser aplicada a distintos obreros directos. Un obrero directo cumple una norma dada en un

determinado por ciento. Un obrero auxiliar realiza el mantenimiento de diferentes equipos y un equipo dado siempre es atendido por un mismo obrero auxiliar. Para cada equipo atendido por un obrero auxiliar dado se conoce el tipo de mantenimiento efectuado y la última fecha en que se realizó. Todos los obreros de la fábrica están caracterizados por un número, su nombre, calificación laboral y categoría ocupacional (directo o auxiliar).

Realice las siguientes actividades:

- a) Identifique las entidades fuertes
- b) Identifique las entidades débiles
- c) Identifique las relaciones entre las entidades
- d) Liste los atributos por cada entidad
- e) Identifique las llaves simples y complejas de todas las entidades
- f) Determine la generalización y agregación del caso
- g) Diseñe el DER final usando yEd.

Caso Propuesto 03: Distribuidora Minerva

Estamos seguros que hoy es su día de suerte, ya que acaba de ser contratado por la empresa Distribuidora “Minerva”, a fin de diseñar su base de datos. El Gerente General le ha dado la siguiente información:

La empresa cuenta con personal en el área de ventas; por ello, se desea almacenar su código, nombre, apellido paterno, apellido materno, sueldo, fecha de ingreso, estado y tipo de personal. En la empresa se cuenta con tres tipos de personal de ventas: los vendedores de campo, los supervisores y los vendedores de oficina. Adicionalmente a los datos ya registrados, de los vendedores de campo se desea registrar el monto de su comisión por ventas, también, de los supervisores, las zonas que tienen a su cargo para supervisar, y de los vendedores de oficina el número, anexo telefónico, en el cual pueden ser ubicados. Se debe de tener en cuenta que los supervisores tienen a su cargo un grupo de vendedores de campo, pero un vendedor de campo solo reporta a un supervisor.

Esta empresa tiene un catálogo amplio de productos, los cuales los clasifica en industriales y domésticos. De los productos en general se requiere almacenar su código, descripción, precio de compra, precio de venta, lugar de procedencia y tipo; sólo de los productos industriales se requiere saber su rubro. Los productos que comercializa esta distribuidora son proveídos por diversas empresas proveedoras de las mismas que se requiere almacenar su código, razón social y dirección. Cabe indicar, que un mismo producto puede ser proveído por diferentes empresas proveedoras. Un requerimiento específico del Gerente General fue saber la situación actual de cada uno de los productos proveídos por cada proveedor.

Eventualmente la empresa desarrolla promociones de ventas, las mismas que son realizadas por un determinado vendedor de campo, de un determinado producto en un determinado distrito; de estas promociones se desea almacenar su fecha de inicio y fecha de fin. De los distritos se desea registrar su código y la descripción de mismo.

Realice las siguientes actividades:

- a) Identifique las entidades fuertes
- b) Identifique las entidades débiles
- c) Identifique las relaciones entre las entidades

- d) Liste los atributos por cada entidad
- e) Identifique las llaves simples y complejas de todas las entidades
- f) Determine la generalización y agregación del caso
- g) Diseñe el DER final usando yEd.

Casos Propuestos adicionales

Realice las siguientes actividades tomando en cuenta lo siguiente:

- a) Identifique las entidades fuertes
- b) Identifique las entidades débiles
- c) Identifique las relaciones entre las entidades
- d) Liste los atributos por cada entidad
- e) Identifique las llaves simples y complejas de todas las entidades
- f) Determine la generalización y agregación del caso
- g) Diseñe el DER final usando yEd.

Caso #1

En una empresa de taxis de Lima existe un conjunto de taxis que brindan servicio a los hospitales de dicha ciudad. En la empresa trabajan choferes, cada uno de los cuales se caracteriza por su DNI, nombre y años de servicio. En la empresa hay numerosos taxis, de los que se conoce la placa, el número del motor, la marca y el modelo. Un taxi puede ser conducido por diferentes choferes (en distintos momentos), pero un chofer siempre conduce el mismo taxi. Se conoce la cantidad de kilómetros totales recorridos por un chofer en su correspondiente taxi. La empresa brinda servicios a varios hospitales. De cada hospital se conoce su nombre, tipo y dirección. Un chofer le brinda servicios (realiza viajes) a distintos hospitales y a un hospital le brindan servicios distintos choferes. Se sabe la cantidad de viajes realizados por cada chofer a cada hospital.

Caso #2

En un organismo se reciben productos que son importados de diferentes países, de modo que un producto puede importarse de varios países y de un país se pueden importar distintos productos. Un producto se importa de un país en cierta cantidad. Es necesario controlar las cantidades que se importan de cada país y el valor de las importaciones. Las propiedades de los productos son los siguientes: número, que lo identifica, nombre, unidad de medida, peso y precio unitario. Las propiedades de los países son los siguientes: número, que lo identifica, nombre, zona geográfica y tipo de moneda.

Caso #3

El Instituto de Aeronáutica del Perú cuenta con información sobre las diferentes compañías de aviación que existen en el mundo. De cada compañía se conoce su nombre (Cubana, Copa, AeroContinente, etc.), su volumen anual de ventas y un código que la identifica. Cada compañía puede estar representada en diferentes países y en un país pueden estar representadas diversas compañías. De cada país se conoce su código, nombre, idioma y área de moneda.

También se sabe que por vía aérea se realizan diferentes viajes. Cada viaje posee un código, un lugar de origen, un destino y una cantidad de kilómetros a recorrer. Además, se conoce que existen distintos tipos de aviones. Cada tipo de avión se identifica por un nombre (IL-62, DC-10, etc.) y posee un consumo de gasolina por kilómetro y una

cierta cantidad de asientos. Un mismo viaje puede ser realizado por distintos tipos de aviones y estos pueden ser utilizados en diversos viajes. Un viaje realizado por un tipo de avión constituye un vuelo y para cada vuelo uno se conoce su tiempo de duración. Una compañía de aviación realiza muchos vuelos y un mismo vuelo puede ser cubierto por diversas compañías. Para cada vuelo que ofrece una compañía se conoce el precio del pasaje.

Caso #4

Se desea controlar la actividad de reparación de radios en los talleres dedicados a esta tarea. Cada usuario puede llevar diferentes radios para ser reparados, aunque un radio sólo pertenece a un usuario. Para cada radio se conoce el número de la solicitud de reparación, que lo identifica, tipo de radio, descripción y tipo de rotura (**sencillas o complejas**). De cada usuario se conoce su DNI, nombre y dirección. Cada técnico de reparaciones pertenece a un taller y tiene un código que lo identifica dentro de su taller, por lo que cada código puede repetirse para diferentes talleres, y tiene, además, un nombre, una categoría (**A, B, C**) y un salario. En un taller laboran muchos técnicos de reparaciones. De cada taller se conoce su código, que lo identifica, nombre y dirección. Los radios con roturas complejas sólo podrán ser atendidos por técnicos de categoría **A**, mientras que los radios con roturas sencillas pueden ser atendidos por cualquier técnico. En cualquier caso, un técnico puede reparar muchos radios, pero un radio es reparado por un solo técnico.

Caso #5

En un taller de mantenimiento de equipos de computación se desea controlar los servicios brindados. En el taller laboran varios trabajadores. De cada uno se conoce su DNI, su nombre, categoría laboral y tipo de mantenimiento (**Tipo1, Tipo2**) que se encarga de realizar a los equipos. El mantenimiento de los equipos de computación se ejecuta por órdenes de servicio que son presentadas por las empresas usuarias de este taller. Una empresa usuaria puede presentar muchas órdenes de servicio y cada orden de servicio es presentada por una empresa usuaria. De cada orden de servicio se conoce su número, que la identifica dentro de la empresa que la presenta, por lo que puede repetirse ese número para diferentes empresas, y la fecha en que se presenta. De cada empresa usuaria se sabe el código que la identifica, su nombre y la rama económica a la que pertenece.

Una orden de servicio puede contener la solicitud de mantenimiento de varios equipos y puede solicitarse el mantenimiento de un equipo en diferentes órdenes de servicio. De cada equipo se sabe el código que lo identifica, su descripción, su marca y su tipo (sólo se toman en cuenta dos tipos: **Computadoras y Otros**). El mantenimiento de las **computadoras** sólo puede ser realizado por los trabajadores que realizan mantenimientos **Tipo1** y el mantenimiento de los **Otros** equipos sólo puede realizarse por trabajadores encargados de hacer los mantenimientos **Tipo2**. En ambos casos, el mantenimiento de un equipo es realizado por un solo trabajador, aunque un trabajador puede realizar el mantenimiento de varios equipos.

Caso #6

Se desea controlar la programación de los itinerarios a recorrer por las locomotoras en una estación de ferrocarriles. En la estación existen varias locomotoras, de las que se conoce el código que la identifica y los años que lleva prestando servicios.

Existen diversas marcas de locomotora; una locomotora es de una marca y una marca puede serlo de varias locomotoras. De cada marca se tiene el código que la identifica, el consumo de combustible promedio por hora y la velocidad máxima que puede alcanzar. Una locomotora realiza diversos itinerarios, pero un itinerario lo realiza una sola locomotora. De cada itinerario se conoce el número que lo identifica para una locomotora dada, pero que puede repetirse para distintas locomotoras (es decir, la locomotora A puede tener un itinerario 2 y la locomotora B puede tener también un itinerario 2), los kilómetros a recorrer, el destino final y el tipo de itinerario (**Diurno** o **Nocturno**; un itinerario es diurno o es nocturno y todos los itinerarios son de uno de estos dos tipos).

Las locomotoras son conducidas por maquinistas. Un maquinista (cualquiera) puede conducir varias locomotoras y una locomotora puede ser conducida por muchos maquinistas y se conoce la cantidad de kilómetros recorrido por un maquinista en una locomotora. De cada maquinista se conoce su DNI, que lo identifica, su nombre y si tiene más de 10 años de antigüedad como maquinista o no. Los itinerarios nocturnos sólo los pueden realizar los maquinistas con más de 10 años de antigüedad y los maquinistas con más de 10 años de antigüedad sólo realizan itinerarios nocturnos. Los itinerarios diurnos sólo los realizan los maquinistas con 10 o menos años de antigüedad y los maquinistas con 10 o menos años de antigüedad sólo realizan itinerarios diurnos. En ambos casos ocurre que un maquinista realiza muchos itinerarios y que un itinerario es realizado por muchos maquinistas.

Casos de aplicaciones propuestos

Realice las siguientes actividades tomando en cuenta lo siguiente:

- a) Identifique las entidades fuertes
- b) Identifique las entidades débiles
- c) Identifique las relaciones entre las entidades
- d) Liste los atributos por cada entidad
- e) Identifique las llaves simples y complejas de todas las entidades
- f) Determine la generalización y agregación del caso
- g) Diseñe el DER final usando yEd.

Aplicación #1: Empresa “200 Millas”

La empresa “200 Millas” se dedica a la extracción y fabricación de productos derivados de la rica fauna marítima de nuestro litoral. Tiene su sede en el Callao y desde allí realiza las actividades de producción y distribución de sus productos a sus diferentes clientes. Los productos son tanto para el consumo humano (Conservas conservas) como también para la industria, como por ejemplo fertilizantes, insumos químicos, etc., y a lo largo de sus 5 años de funcionamiento tiene una gran acogida debido a la calidad de productos que fabrica, así como los precios con los cuales los comercializa.

En el último año, el Ing. Juan Quiroz, gerente general de “200 millas” ha decidido que para seguir con el ritmo de crecimiento de la empresa y convertirla en un corto plazo en una empresa líder dentro del ámbito donde se desarrolla, se necesita ser cada vez más competitivos, lo que implica tener un mejor control sobre las materias primas, el proceso productivo, la distribución de los productos, su comercialización, etc. Se ha elaborado para ello una cartera de proyectos, de los cuales algunos están en su fase de análisis y otros en la de desarrollo.

Dentro de estos últimos se encuentra el Sistema de Control del Proceso Productivo, el cual tiene un módulo denominado “Seguimiento de Extracción de Materia Prima”. Éste controla las diferentes actividades desde la extracción de la materia prima hasta la fabricación de los productos finales. El módulo permitirá efectuar un adecuado seguimiento y registro de toda la información pertinente. El responsable del proyecto es el Sr. analista Rubén Díaz, quien actualmente se encuentra en la etapa del diseño de una base de datos la cual permita el adecuado registro de toda la información derivada de las actividades propias del módulo.

En sus constantes visitas y entrevistas con la Ing. Inés Tang, supervisora del Área de Producción, Ella ha manifestado lo siguiente: “La empresa posee una flota de aproximadamente 100 embarcaciones de distinta capacidad y tipo. Es importante para nosotros almacenar la información de cada una de las embarcaciones, las cuales están debidamente codificadas. Es necesario registrar también el nombre, fecha de compra, tipo de embarcación y capacidad de almacenamiento.” En las diferentes embarcaciones se encuentran trabajando personal debidamente capacitado en la ubicación y extracción del pescado. Es de nuestra incumbencia mantener un catálogo actualizado de dicho personal, en el cual se registren el código, apellido paterno, apellido materno, nombres, fecha de nacimiento, fecha de ingreso, libreta electoral, cargo, haber básico y fecha de ultima capacitación. Un trabajador solo está asignado a una embarcación y para llevar el manejo de una embarcación se requiere la participación de varios trabajadores.

Las actividades de pesca son de gran importancia dentro del proceso productivo. Nosotros las llamamos faenas de pesca. Estas son ejecutadas por una determinada embarcación, por lo que es lógico pensar que una embarcación ha ejecutado varias faenas de pesca. Cada una tiene un número que nos permite identificarla y distinguirla de otras. Es necesario mantener información de cada faena almacenando su número, fecha de inicio, fecha de fin, zona del mar donde se ejecutó, así como la embarcación que la ejecutó.

Para un mejor control de las especies marinas que extraemos de nuestro litoral se maneja un catálogo denominado Especies, en el cual se detalla la información más relevante de las materias primas (pescados) que permiten la elaboración de nuestros productos. En este catálogo se registra información como el código del pescado, nombre del pescado, tipo de pescado y su cotización en el mercado (por tonelada métrica).

Se sabe que una faena de pesca puede recoger una o más especies, por ejemplo, la Faena “F093” recogió atún, sardina y anchoveta y, como es lógico deducir, una especie puede ser recogida en varias faenas.

Por último es importante almacenar la información acerca de los diferentes productos que se fabrican en “200 Millas”. Para tal efecto se maneja una relación con la información pertinente de cada producto, como su código, descripción, tipo, precio de venta y unidad de medida. Para la fabricación de un determinado producto se emplean una o más especies, y una especie sirve para la preparación de diferentes productos.

Finalmente, necesitamos una base de datos que nos permita almacenar la información descrita anteriormente, así como satisfacer necesidades puntuales como las siguientes:

- ¿Qué barco ejecutó una determinada faena?
- ¿Qué especies y en qué cantidad se recogió en una determinada faena?

- ¿Qué especies y en qué cantidad se necesitan para la elaboración de un determinado producto?
- ¿Qué empleados laboran en un barco, entre otros requerimientos de información?

La elaboración de una adecuada estructura de base de datos garantizará la satisfacción de las necesidades establecidas.

Tras el manifiesto de la Srta. Tang, el Sr. Díaz debe de elaborar la base de datos que permita el soporte de manejo de información para el módulo a desarrollar. Asuma Ud. el papel del Sr. Díaz y elabore los modelos genéricos y detallados , así como las consideraciones finales necesarias para la implementación de la base de datos en cuestión.

Aplicación #2: Comercial “Santa Úrsula”

La empresa Comercial “**Santa Úrsula**” está ubicada en La Victoria y se dedica a la comercialización de diversos productos para seguridad industrial, útiles de oficina, insumos de cómputo y de limpieza en general.

Se ha encomendado al Ing. Juan Mendoza la elaboración de una base de datos que permita controlar las transacciones más importantes de la empresa, de tal forma que pueda dar soporte de datos a las diferentes aplicaciones que en el futuro se desean implementar. Se ha decidido iniciar el proyecto con el Módulo de Ventas y Abastecimiento por lo que se requiere de identificar las principales entidades que son de interés en estas áreas. Tras una serie de reuniones con los usuarios y sus colaboradores, el Ing. Mendoza ha llegado a las siguientes conclusiones: Cada producto posee un código que los distingue además de su descripción, precio actual, unidad de medida, línea (industrial, oficina, cómputo o limpieza), stock actual, stock mínimo y un campo que permite saber si el producto es o no importado.

Los clientes a los que se atienden son todas empresas de las cuales se registra su código, razón social, dirección, teléfono, RUC, email y el nombre del contacto.

Los clientes pueden generar varias facturas, pero cada una de estas sale a nombre de un determinado cliente.

Los vendedores son los que generan las ventas y se les registra en cada factura que se genera. Operan en varios distritos, pero en un distrito solo se asigna a un vendedor.

Existen también vendedores de mostrador, que no se asignan a ningún distrito y trabajan en la sede principal. Se debe registrar el código del vendedor, así como su nombre, apellido, fecha de ingreso, haber básico, y su tipo (estable o contratado).

Las transacciones de venta que se realizan con estas empresas se registran a través de facturas, en las cuales a su vez pueden contener uno o más productos. A su vez, un producto se comercializa en varias facturas. Las facturas tienen un número de factura, fecha de emisión, un estado (pendiente, cancelada o anulada) y el porcentaje de IGV de la factura. Se sabe que un mismo cliente puede generar una o más facturas. Cada una es dirigida a un cliente y sale a nombre del vendedor que originó la transacción, por lo que es lógico pensar que un vendedor ha originado varias facturas.

También se maneja información sobre proveedores. Están registrados en un catálogo donde figuran su código, razón social, dirección, teléfono, RUC y email. Además, como siempre tengo que estar con productos en stock, manejo la posibilidad de que un

proveedor me abastezca de uno o más productos y viceversa, los cuales me lo abastecen a diferentes precios.

Las transacciones de compra se registran en órdenes de compra, en las cuales se solicita el abastecimiento de uno o más productos. Por supuesto, cada vez que se necesita de un producto, este se solicita mediante la orden de compra respectiva. Se maneja un número de orden de compra, fecha de emisión y su estado (pendiente, atendida o rechazada). Un proveedor puede atender una o más órdenes de compra y cada orden se emite a un proveedor.

Como deseo ubicar estratégicamente a mi clientela, vendedores y proveedores, tengo un catálogo de distritos donde almaceno el código del distrito y su nombre. Es decir, en un distrito ubico a uno o más clientes y proveedores. Un vendedor se asigna a varios distritos y en un distrito opera un solovendedor.

La base de datos a desarrollar debe permitir realizar, entre otras, las siguientes consultas:

- ¿Cuál es el volumen de facturas por cliente?
- ¿Qué artículo es más solicitado por distrito?
- ¿Qué artículos se han comercializado en cada factura, cantidad y precio?
- ¿Qué alternativas tengo cuando un proveedor no tiene el producto que se necesita?
- ¿El vendedor que más facturas ha generado?
- ¿Cuál es la línea de productos que más demanda tiene?

Asuma el papel del Ing. Mendoza y elabore un modelo entidad- relación genérica y detallada para el caso antes descrito.

Aplicación #3: “Cyber Ventas”

La empresa **Cyber** Construcción se dedica a la construcción de diferentes obras que son parte de proyectos sociales, en los cuales se emplean máquinas de propiedad de la empresa o alquiladas a terceros. En cada proyecto, se asignan diferentes tipos de personal como jefes de proyecto y los empleados (obreros).

La empresa cuenta con una serie de máquinas de las cuales se almacena el código, fecha de compra, tipo y horas trabajadas. Las máquinas se dividen en propias y alquiladas.

Cuando una obra no puede ser atendida con maquinaria de la empresa se contrata los servicios de un tercero, al cual se le denomina Team Externo. Éste se conforma por máquinas del tipo alquiladas que cumplen una función y son proporcionadas por un proveedor.

Un proveedor provee muchas máquinas alquiladas que cumplen una determinada Función. Ésta es cumplida por muchas máquinas alquiladas que son proporcionadas por un proveedor. Un equipo alquilado es proporcionado por un proveedor y cumple una determinada función. De los Team externos que se conforman se almacena un código autogenerado y una fecha en la que se conforman. De los proveedores se almacena su RUC, razón social, dirección y teléfono. Para la función que cumplen las máquinas alquiladas se cuenta con un código de la misma.

La atención de una obra puede ser de varias máquinas propias pero estas son asignadas a una determinada obra por un período de tiempo. Sin embargo, algunas obras requieren de uno o varios Team externos dependiendo de la magnitud de la misma. Se almacena un número autogenerado, la fecha de inicio y término, y el monto de la misma.

Las obras que se realizan son parte de proyectos, de tal forma que un proyecto puede involucrar varias obras. Para los proyectos se almacena un número, fecha de inicio y término y el número de etapas.

A cada proyecto se le asignan varios trabajadores (que en la empresa se les denomina proyecto) de los cuales se almacena su código, DNI, nombres, apellidos, dirección, teléfono domiciliario, fecha de ingreso a la empresa y su cargo. Es importante precisar que un empleado (Personal) puede ser el jefe de otros empleados, al que se le considera Jefe del Proyecto.

Sobre la base del caso planteado se pide que:

1. Indique si las siguientes acciones se pueden realizar y justifique su respuesta:
 - a. ¿Es posible actualizar los datos de un proveedor sin afectar la composición del Team Externo?
 - b. El Gerente de Logística desea ingresar las características de nuevas máquinas del tipo alquiladas.
 - c. Debido a que la nueva política de la empresa, es no conservar información pasada, el Gerente de Logística ha dispuesto que Sistemas elimine todos los Team externos que no han sido considerados desde el año anterior para una obra.

“La educación es el arma más poderosa que puedes usar para cambiar el mundo”

Nelson Mandela

Resumen

- El Modelo entidad-relación permite representar lógicamente un caso.
- En el diagrama entidad-relación, una entidad se representa mediante un rectángulo, una relación mediante un rombo, un atributo mediante un círculo. Todos estos elementos deben aparecer debidamente identificados por medio de un nombre. Los atributos que constituyen la llave de una entidad, deben tener el círculo relleno.
- Una entidad cuyos atributos no sean suficientes para identificarla se denomina débil y su llave está formada por algún o algunos de sus atributos más la llave de la entidad que le da origen. Se representa con un doble rectángulo y con la relación entre ella y la entidad que le da origen. Esta relación es de, a lo sumo, muchos (por el extremo de la débil) a uno (por el extremo de la entidad que la origina).
- La llave de una relación de **m:n** está formada por la llave de las entidades que participan en la relación. La llave de una relación de **1:n** está formada por la llave de la entidad del extremo muchos. La llave de una relación de **1:1** está formada por la llave de cualquiera de las entidades que participan.
- En una Generalización/Especialización, la entidad generalizada describe las características generales o comunes que son aplicables a todas las especializaciones. Las especializaciones, como casos especiales de la generalización, sólo contemplan sus propiedades particulares. La llave de cada Especialización es la misma de la generalización.
- Una agregación es el resultado de considerar una relación como una entidad. Los atributos de la relación pasan a ser atributos de la entidad agregada. La llave de la entidad agregada es la llave de la relación que la originó, excepto en el caso en que se defina especialmente un identificador para la agregación, pasando entonces a ser la llave el identificador.
- Todos estos elementos pueden combinarse en un DER. Por ejemplo, una entidad débil puede ser, a su vez, una generalización que tenga sus especializaciones; una generalización puede tener especializaciones que, a su vez, puedan ser generalizaciones de otras especializaciones; entre las entidades que participan en una agregación puede haber una entidad débil, etc.

Si desea saber más acerca de estos temas, puede consultar las siguientes páginas.

Link	Descripción
http://www.ongei.gob.pe/publica/metodologias/Lib5011/cap2-3.htm	En esta página web hallará algunos conceptos complementarios a los mostrados en el manual sobre modelo conceptual de datos.
http://elies.rediris.es/elies9/5-2.htm	En esta página web encontrará definiciones complementarias al modelo conceptual.
http://www.ecured.cu/index.php/Diagrama_Entidad_Relaci%C3%B3n	En esta página web hallará algunos conceptos complementarios al diagrama entidad relación
http://jorgesanchez.net/manuales/gbd/entity-relacion.html	En esta página web hallará algunos conceptos complementarios al diagrama entidad relación



MODELO RELACIONAL. NORMALIZACIÓN

LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la tercera unidad, el estudiante organiza datos no estructurados y los integra al diseño de la base de datos a partir de documentos comerciales utilizando la técnica de Normalización (1FN, 2FN, 3FN).

TEMARIO

3.1. Tema 7 : Normalización parte I

- 3.1.1 : Modelo relacional
- 3.1.2 : Primera Forma Normal
- 3.1.3 : Segunda Forma Normal
- 3.1.4 : Tercera Forma Normal
- 3.1.5 : Integración: Ejercicios propuestos
- 3.1.6 : Ejercicios de normalización básicos

3.2. Tema 8 : Normalización parte II

- 3.2.1 : Sin grupos repetitivos
- 3.2.2 : Con un grupo repetitivo
- 3.2.3 : Ejercicios
- 3.2.4 : Múltiples grupos repetitivos
- 3.2.5 : Ejercicios
- 3.2.6 : **Diccionario de datos**

3.3. Tema 9 : Normalización parte III

- 3.3.1 : Normalización Detalle de Detalle / Ítems
- 3.3.2 : Ejercicios parte I
- 3.3.3 : Ejercicios propuestos: Detalle del Detalle / Ítems Detalle
- 3.3.4 : Ejercicios propuestos (adicionales)

ACTIVIDADES PROPUESTAS

1. Explicaciones por parte del facilitador combinadas con ejemplos.
2. Se efectúan preguntas de comprobación dirigidas a los alumnos sobre los temas tratados.

3.1. TEMA 7: NORMALIZACIÓN PARTE I

3.1.1. Modelo Relacional

Uno de los modelos matemáticos más importantes y actuales para la representación de las bases de datos, es el enfoque relacional. Este se basa en la teoría matemática de las relaciones lo que le suministra una fundamentación teórica que permite aplicar todos los resultados de dicha teoría a problemas tales como el diseño de sublenguajes de datos y otros.

El término relación se puede definir matemáticamente como sigue:

- Dados los conjuntos D_1, D, \dots, D_n (no necesariamente distintos), R es una relación sobre esos n conjuntos si está constituida por un conjunto de n -tuplos ordenados d_1, d_2, \dots, d_n tales que $d_1 \in D_1, d_2 \in D_2, \dots, d_n \in D_n$.
- Los conjuntos D_1, D_2, \dots, D_n se llaman dominios de R y n constituye el grado de la relación. La cantidad de tuplas constituye la cardinalidad (tipo de relación de correspondencia 1-1, 1-n, n-1 y n-n)

Es conveniente representar una relación como una tabla bidimensional donde cada fila representa un n -tuple, tal como se muestra en la siguiente imagen:

COLUMNA1	COLUMNA2	COLUMNA3	COLUMNA4	
CODIGO DE FACTURA	FECHA FACTURADA	CODIGO DE VENDEDOR	CODIGO DE CLIENTE	
FAC001	2015-04-05 10:13:39.497	VEN001	CLI001	FILA 1
FAC002	2015-04-05 10:13:39.497	VEN001	CLI002	FILA 2

Las columnas de la tabla también son consideradas como **atributos** de una entidad y las filas como **ocurrencias** o tuplas.

En el modelo relacional, tanto los objetos o entidades, como las relaciones que se establecen entre ellos, se representan a través de "tablas", que en la terminología relacional se denominan relaciones.

Cada relación está compuesta de filas (las ocurrencias de los objetos) y se les denomina, en la terminología relacional, como tuplos, tuplas o uplos, uplas (en realidad, n -tuplos, pero en muchos casos se suprime la n cuando no existe posibilidad de confusión).

También la relación está compuesta por columnas (los atributos o campos que toman valores en sus respectivos dominios).

Debemos tener en cuenta los siguientes aspectos:

1. No puede haber dos filas o tuplas iguales en una misma tabla o entidad.
2. El orden asignado a las filas no es significativo.
3. El orden asignado a las columnas no es significativo.
4. Cada valor dentro de la relación (cada valor de un atributo) es un dato atómico (o elemental), es decir, no descomponible; por ejemplo, un número, una cadena de caracteres. En otras palabras, en cada posición fila-columna existe un solo valor, nunca un conjunto de valores.

Una relación que satisface este último punto se denomina "**normalizada**", aunque veremos más adelante que, en realidad, lo que ocurre es que está en Primera Forma Normal.

La teoría de la **normalización** se basa en la necesidad de encontrar una representación del conjunto de relaciones que en el proceso de actualización sea más adecuada. Llevar una relación no normalizada a normalizada es muy simple.

Existen diferentes niveles de normalización que se llaman formas normales que veremos más adelante. Veamos un ejemplo donde un suministrador y producto se puede representar fácil y claramente mediante el modelo relacional.

Los atributos de estas dos entidades son los siguientes:

- **Suministrador:** Número, que lo identifica, Nombre, Tipo y Distrito donde radica.
- **Producto:** Número, que lo identifica, Nombre, Precio unitario y Peso.

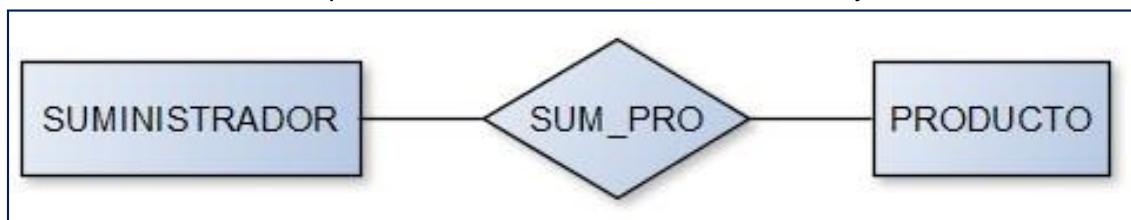


Figura 19: Modelo Entidad Relación
Fuente. - Tomado desde la aplicación yEd

Además, un **suministrador** puede suministrar muchos **productos** y un **producto** puede ser suministrado por varios **suministradores**. Se conoce la cantidad de un determinado producto que suministra un suministrador dado.

TABLA: **SUMINISTRADOR**

CÓDIGO_SUM	NOMBRE_SUM	TIPO_SUM	DISTRITO_SUM
S001	Maria Rojas Zamora	30	San Isidro
S002	Carla Segura Ramos	10	Miraflores
S003	Jenifer Lopez Arias	20	Lince
S004	Guadalupe Salvatierra Gonzales	20	Lince
S005	Janeth Cruz Cruz	15	Surco

TABLA: **PRODUCTO**

CÓDIGO_PRO	DESCRIP_PRO	PRECIO_PRO	PESO_PRO
P001	Clavo 1 pulgada.	S/. 0.20	12
P002	Tuerca 3/8	S/. 0.50	17
P003	Martillo	S/. 15.50	80
P004	Tornillo 1 1/2	S/. 2.00	10
P005	Alicate de presión	S/. 14.00	50

TABLA: SUM-PRO		
CÓDIGO_SUM	CÓDIGO_PRO	CANTIDAD
S001	P003	100
S002	P001	150
S002	P003	120
S005	P002	50
S004	P002	30
S001	P005	50
S003	P003	20
S001	P001	10
S002	P003	15
S005	P005	2
S001	P003	50
S002	P002	60
S001	P001	10
S004	P003	14

La representación en el modelo relacional es más simple que con el modelo jerárquico y el modelo reticular, ya que con tres (3) tablas se tiene todo el modelo representado. En el **modelo relacional**, el resultado de una demanda es también una relación y las demandas simétricas (en el sentido de ser una la inversa de la otra; por ejemplo:

- Si queremos recuperar los números de los suministradores que suministran el producto "P4".
- Y recuperar los números de los productos que suministra el suministrador "S2" requieren operaciones simétricas.

Las diversas formas de expresar las recuperaciones dan lugar a los lenguajes relacionales cuyas formas más representativas son:

- **Álgebra relacional** (basado en las operaciones del álgebra de relaciones)
- **Cálculo relacional** (basado en el cálculo de predicados)

Ventajas del modelo relacional:

- Una de las principales ventajas es su simplicidad, pues el usuario formula sus demandas en términos del contenido informativo de la base de datos sin tener que atender a las complejidades de la realización del sistema, lo que implica gran independencia de los datos.
- La información se maneja en forma de tablas, lo que constituye una manera familiar de representarla.
- Al igual que en el modelo reticular, si se tienen relaciones normalizadas, no surgen dificultades grandes en la actualización.

Veamos en el modelo del **SUMINISTRADOR-PRODUCTO** presentado anteriormente, un ejemplo de cada tipo de operación de actualización:

- **Creación.** - Añadir un producto P7. Se agrega la nueva ocurrencia en la tabla PRODUCTO. Es posible hacerlo, aunque ningún suministrador lo suministre.

- **Eliminación.** - Se puede eliminar el suministrador S1 sin perder el producto P6, a pesar de que es el único suministrador que lo suministra.
- **Modificación.** - Se puede cambiar el precio del producto P2 sin necesidad de búsquedas adicionales ni posibilidad de inconsistencias.

No obstante, veremos que el proceso de normalización no es suficiente hasta el punto aquí visto.

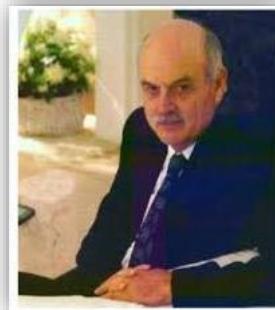
Desventajas

Se dice que la fundamental desventaja consiste en la dificultad de lograr productividad adecuada de los sistemas, ya que no se emplean los medios técnicos idóneos, tales como las memorias asociativas, por ello, es siendo necesario simular este proceso, pero, en realidad, la eficiencia y productividad de los sistemas actuales resultan realmente muy satisfactorias.

Normalización

La teoría de la normalización se ha desarrollado para obtener estructuras de datos eficientes que eviten las anomalías de actualización. El concepto de normalización fue introducido por **E.F. CODD** y fue pensado para aplicarse a sistemas relacionales. Sin embargo, tiene aplicaciones más amplias.

La normalización es la expresión formal del modo de realizar un buen diseño. Provee los medios necesarios para describir la estructura lógica de los datos en un sistema de información.



Ventajas de la normalización

- Evita anomalías en la actualización.
- Mejora la independencia de los datos, porque permite realizar extensiones de la base de datos, afectando muy poco, o nada, a los programas de aplicación existentes que accedan la base de datos.

La normalización involucra varias fases que se realizan en orden. La realización de la 2º fase supone que se ha concluido la 1ra. Tras completar cada fase se dice que la relación está en:

- Primera Forma Normal (1FN)
- Segunda Forma Normal (2FN)
- Tercera Forma Normal (3FN)
- Forma Normal de Boyce-Codd (FNBC)

Existen, además, la Cuarta (4FN) y la Quinta (5FN) Formas Normales.

Las relaciones en 1FN son un subconjunto del universo de todas las relaciones posibles. Las relaciones en 2FN son un subconjunto de las que están en 1FN y así sucesivamente, como se muestra en la siguiente figura:

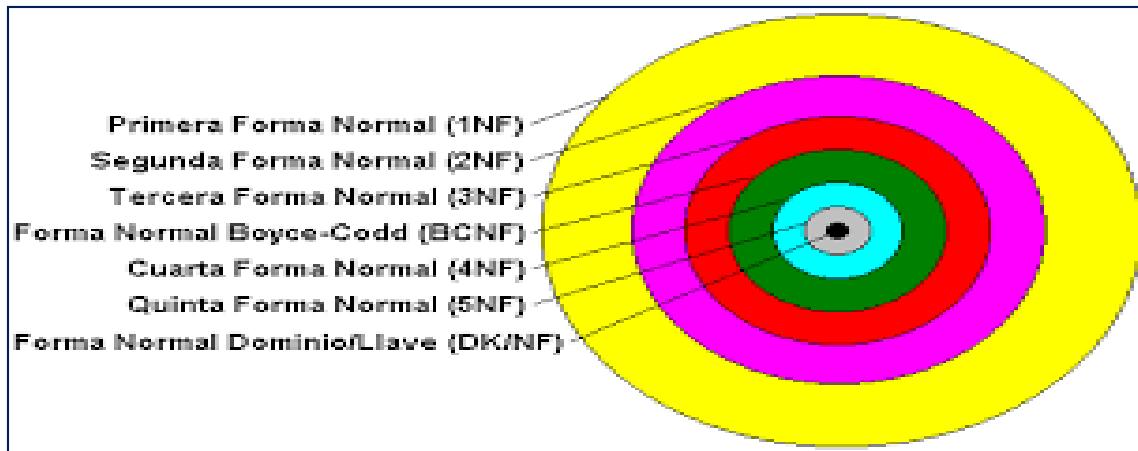


Figura 20: Formas Normales
Fuente.- Tomado de <http://cvitae.tripod.com/loida/normaliz.html>

Definiciones previas

- **Datos atómicos**

Se le llama así cuando el campo de cada registro no posee partes más pequeñas a las que se pueda acceder; es decir solo tiene una ocurrencia, por ejemplo:

ALUMNO					
	codigo	nombre	ap_paterno	ap_materno	fecha_nac
1	ALU001	Jose Luis	Roca	Salas	10/12/1980
2	ALU002	Ana	Urbina	Lozada	11/12/1980
3	ALU003	Jorge	Palacios	Perex	12/12/1980
4	ALU004	Fiorella	Andia	Quispe	13/12/1980

↑
Datos atómicos

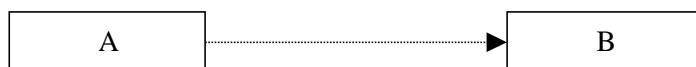
- **Datos no atómicos**

Se le llama así cuando el campo de cada registro tiene más de una ocurrencia, por ejemplo:

ALUMNO									
	A	A	A	A	A	NA	NA	NA	
	codigo	nombre	ap_paterno	ap_materno	fecha_nac	Hobbie	Telefono	Email	
1	ALU001	Jose Luis	Roca	Salas	10/12/1980	Mangas Deporte Lectura	985521218 985522148 5223625	iroca@hotmail.com joroca@peru.com roca_salas@usa.net	
2	ALU002	Carla	Rodriguez	Serna	21/05/1989	Baile Cine Internet Deporte Lectura	965522204	crodiguez@peru.com	
ANOMALIA									

- **Dependencia funcional (DF)**

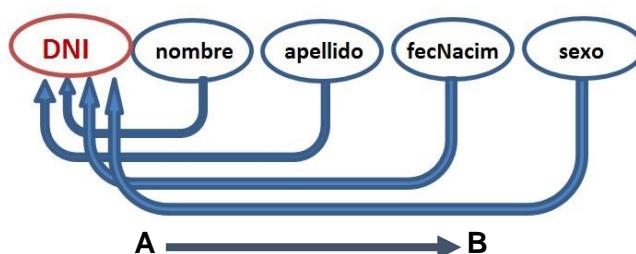
Un atributo **B** de un esquema de relación **R** depende funcionalmente de un atributo **A** de **R**, si y sólo si, cada valor de **A** está asociado con un único valor de **B**. Es decir, dado un valor de **A** queda únicamente determinado el valor de **B**. Se dice que **B** depende funcionalmente de **A**, y que **A** determina funcionalmente a **B**. Tanto **A** como **B** pueden ser atributos simples o compuestos.



DETERMINANTE

Atributo (simple o compuesto) que determina funcionalmente a otros.

Ejemplo:

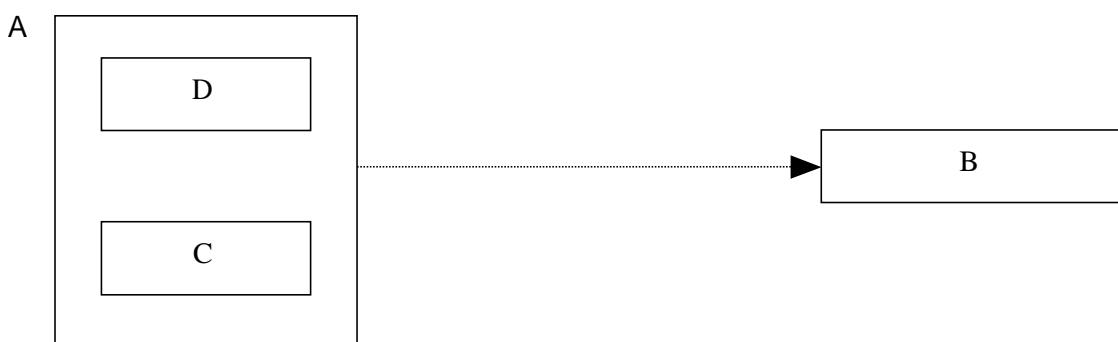


B depende funcionalmente de **A** se denota como:

DNI	Nombre	Apellido	FecNacim	Sexo
40138798	Betsabé	Villanueva Herrera	01/10/1982	F
35255657	Aneth	Veliz Villanueva	31/10/2013	F
56895697	Ruth	Ramos Condori	03/05/1990	F

- **Dependencia funcional (DFC)**

Un atributo **B** de **R** tiene dependencia funcional completa de un atributo **A** de **R**, si tiene dependencia funcional de **A**, pero no tiene dependencia funcional de ningún subconjunto de **A**.



Ejemplo:

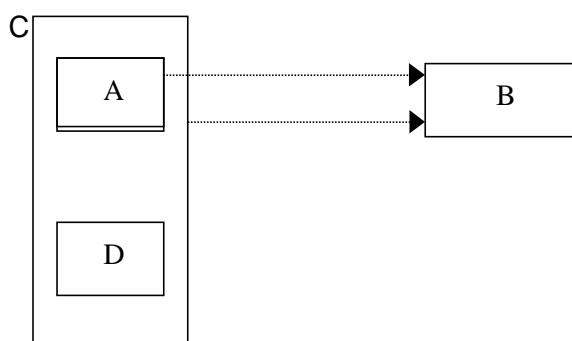
C-Consultor	C-Proyecto	N-Consultor	N-Proyecto	Q Horas trabajadas
C1	P1	Juan	Auditoria	25
C1	P2	Juan	DW	80
C2	P1	Pedro	Auditoria	35
C3	P3	María	CRM	20
C3	P4	María	ERP	50

C-Consultor C-Proyecto Q Horas_Trabajadas

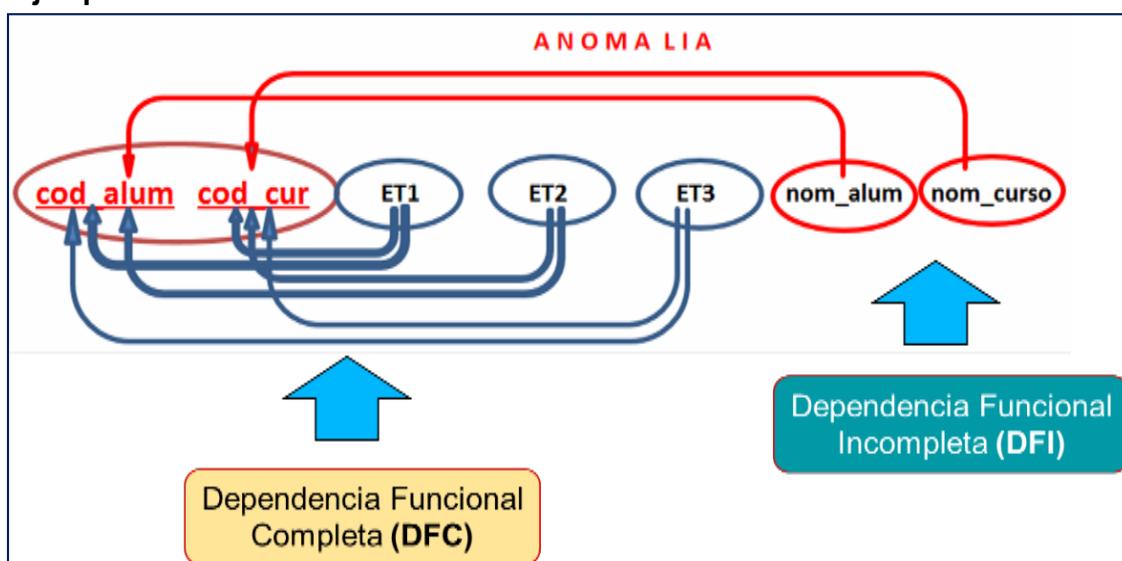
$(C\text{-Consultor}, C\text{-Proyecto}) \rightarrow Q\text{ Horas}_\text{Trabajadas}$

- **Dependencia funcional parcial (DFP)**

Un atributo **B** de **R** tiene dependencia funcional parcial de un atributo **C** de **R**, si tiene dependencia funcional de **C** y además tiene dependencia funcional de un subconjunto propio **A** de **C**.

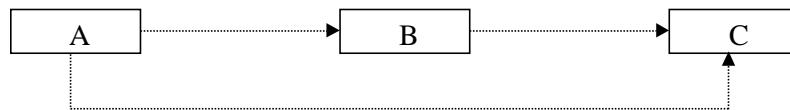


Ejemplo:



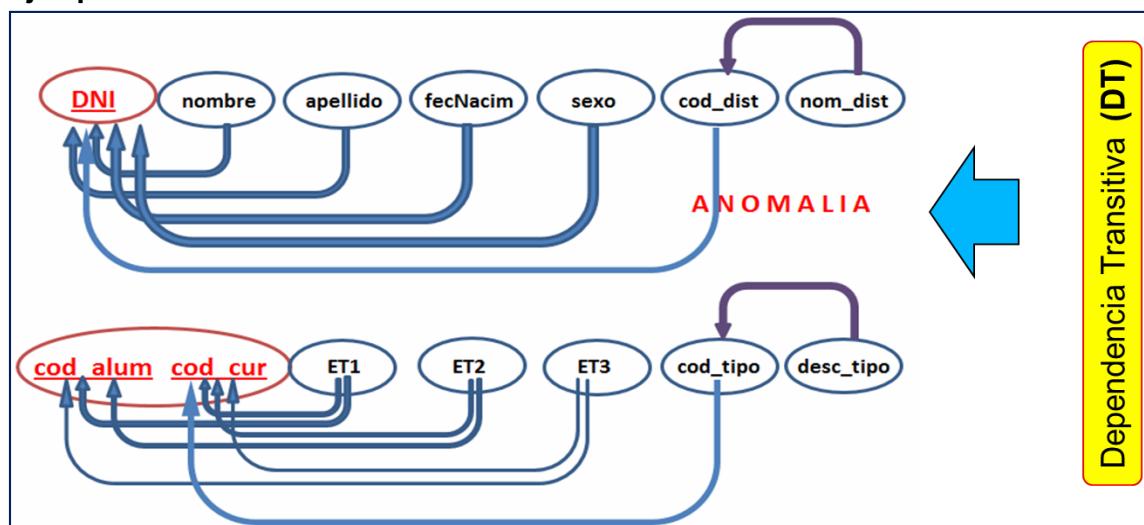
- **Dependencia funcional transitiva (DFT)**

Sean **A**, **B** y **C** atributos de un esquema de relación **R**; si **C** tiene dependencia funcional de **B** y **B** tiene dependencia funcional de **A**, entonces **C** tiene dependencia funcional transitiva de **A**.



Este paso se ejecuta examinando todas las relaciones para ver si hay atributos no llaves que dependan unos de otros. Si se encuentran, se forma una nueva relación para ellos.

Ejemplo:



3.1.2. Primera Forma Normal

Para las explicaciones de los contenidos correspondientes a las formas normales de la 1FN hasta la 3FN, desarrollaremos un ejemplo que consiste en el diseño de la base de datos para la automatización del control de los pedidos de productos y que se presenta a continuación. Supongamos que los modelos para pedir los productos sean como se muestra en la siguiente figura:

FECHA 30/06/2015 28/01/2017	Nº PEDIDO : 02368-52 Nº CLIENTE : 1542 NOMBRE DEL CLIENTE : JUAN JOSE PEREZ LOZA DIRECCION : AV. EL SOL 574 - CHORRILLOS			
DETALLE DEL PEDIDO				
CODIGO DE PRODUCTO	DESCRIPCION	PRECIO UNITARIO	CANTIDAD	TOTAL
15-5562	Televisor 42 pulg. LG	S/. 2,500.00	1	S/. 2,500.00
15-2154	Blue Ray LG	S/. 250.00	1	S/. 250.00
15-0125	Play Station 4 - 500GB	S/. 1,850.00	1	S/. 1,850.00
15-1545	Audifonos Phillips	S/. 50.00	1	S/. 50.00
				IMPORTE TOTAL S/. 4,650.00

Figura 21: Hoja de Pedido
 Fuente. - Tomado desde la aplicación Excel

Listando todos los atributos y se determinar la llave de toda relación.

- a) El análisis de este modelo de pedido de productos muestra que los atributos que se listan a continuación son de interés:

Atributo	Descripción
NUM_PED	Número de pedido.
FEC_PED	Fecha en que se realizó el pedido.
NOM_CLI	Nombre del cliente solicitante.
DIR_CLI	Dirección del cliente.
NUM_PRO	Número o código del producto.
DES_PRO	Descripción del producto solicitado en pedido.
PUNI_PRO	Precio unitario del producto.
CANT_PED	Cantidad de productos solicitados en el pedido.
TOT_PED	Importe por cada producto solicitado.
IMP_PED	Importe total a pagar por todos los productos pedidos.

- b) El **número de pedido** es único y se utiliza para referirse a un pedido, por tanto, se usará como clave (llave).

Nota. - Esta es una suposición inicial, que luego se discutirá en la siguiente sesión.

- c) Se determina los subconjuntos posibles, es decir los valores cuyos datos son repetidos. Aquí debemos mencionar que estos atributos se han determinado observando la boleta de pedido.

Atributo	Descripción
NUM_PRO	Número o código del producto.
DES_PRO	Descripción del producto solicitado en pedido.
PUNI_PRO	Precio unitario del producto.
CANT_PED	Cantidad de productos solicitados en el pedido.
TOT_PED	Importe por cada producto solicitado.

- d) Se determina el atributo del cual dependen los demás atributos total o parcialmente, en este caso es **NUM_PRO**, para crear la clave compuesta.

- e) Luego, se determina la relación resultante:

PEDIDO (**NUM_PED**, FEC_PED, NUM_CLI, NOM_CLI, DIR_CLI,
NUM_PRO, DES_PRO, PUNI_PRO,
CANT_PED, TOT_PED, IMP_PED)

3.1.2.1. Definición de la Primera Forma Normal (1FN)

Definición Principal

“Una relación está en 1FN si cumple la propiedad de que sus dominios no tienen elementos que, a su vez, sean conjuntos”.

Los valores que puede tomar un atributo de una relación son los elementos contenidos en su correspondiente dominio.

Si se permitiera que un elemento del dominio de un atributo fuera un conjunto, entonces, dicho atributo pudiera tomar como valor ese conjunto de valores. Eso

implicaría que en una posición fila, columna habría un conjunto de valores y no un único valor, como hemos visto que debe ocurrir en el modelo relacional.

Segunda definición

“Toda relación normalizada, o sea, con valores atómicos de los atributos, está en 1FN”.

Tercera definición

“Una relación está en 1FN si no incluye ningún grupo repetitivo”.

Del caso se puede observar que la relación **PEDIDO-PRODUCTOS** tiene 5 atributos repetitivos:

NUM_PRO
DES_PRO
PUNI_PRO
CANT_PED

Ya que un pedido puede contener más de una línea de pedido y, por lo tanto, puede contener varios números de producto (NUM_PRO), varias descripciones de producto (DES_PROD), varios precios unitarios (PUNI_PRO) y varias cantidades (CANT_PED). Esta situación acarrea problemas de actualización, pues habría que decidir la cantidad máxima de líneas de pedido de productos que se permitiría en un pedido, dado que los campos de la tabla PEDIDO deben tener un tamaño dado y, entonces, serían capaces de almacenar sólo una determinada cantidad máxima de valores cada uno de ellos.

Esto sería agregar una limitación en el modelo, pues no tiene ese comportamiento en la realidad. Además, en general, se desaprovecharía memoria, dado que, si se decide, por ejemplo, que se va a permitir hasta 20 líneas de pedido en cada pedido, habría que definir tamaños de campos para los grupos repetitivos que permitieran almacenar esa cantidad de valores. Entonces, si en un pedido se solicitan menos de 20 productos, lo cual puede ser muy frecuente, no se utilizaría parte del espacio reservado para cada campo.

Determinar las relaciones de grupos repetidos de los que no lo son.

Hay que eliminar esos grupos repetitivos para que la relación esté en **1FN**. Para ello, se crea:

- 1) Una relación para los campos que sean únicos, es decir, se dejan en la relación original sólo los atributos que no son repetitivos:

PEDIDO (NUM_PED, FEC_PED, NUM_CLI, NOM_CLI, DIR_CLI, PR_PED)

- 2) Se crea una relación para los grupos repetitivos. Además, se crea una llave compuesta formada por la llave primaria de la relación original (NUM_PED) y el atributo del cual dependen los demás atributos repetidos total o parcialmente, en este caso es **NUM_PROD**:

PED-PROD (NUM_PED, NUM_PROD, DES_PRO, PUNI_PRO, CANT_PED, TOT_PED)

Determinación de la llave de cada relación.

Ambas tienen como llave, o parte de la llave, a **NUM_PED**. Pero en **PED-PROD** es necesario la llave compuesta para identificar los productos individuales pues, por ejemplo, **CANT_PED** se refiere a la cantidad en que se pide un determinado producto en un pedido dado (el producto puede solicitarse en otros pedidos en diferentes

cantidades y en el pedido se pueden estar pidiendo otros productos en diferentes cantidades) y, por lo tanto, para identificar una cantidad dada es necesario referirse al pedido y al producto correspondiente.

Ahora, estas nuevas dos relaciones en 1FN modelan el que nos ocupa. Los problemas de actualización mencionados anteriormente quedan resueltos con este nuevo modelo. En lugar de tener varios valores en cada campo de acuerdo a la cantidad de líneas de pedido, tal y como ocurría en la tabla **PEDIDO** original, se tienen varias ocurrencias en la tabla **PED-PROD**, una por cada producto que se solicita en el pedido. Esto permite que se soliciten tantos productos como se desee en cada pedido, pues sólo significa agregar una nueva ocurrencia en la relación **PED-PROD** por cada producto solicitado. Sin embargo, este modelo en 1FN tiene aún problemas de actualización, como se muestra en las siguientes operaciones:

- **Creación.** - La información sobre un nuevo producto no se puede insertar si no hay un pedido que lo incluya.
- **Eliminación.** - Eliminar una línea de pedido que sea la única que pida un producto implica perder la información del producto.
- **Modificación.** - Por cada línea de pedido en la que se solicite determinado producto se tiene una ocurrencia en PED-PROD, que repite la información sobre éste. Si cambia algún atributo del producto, entonces, es necesario hacer muchas actualizaciones.

Entonces, será necesario aplicar formas normales más fuertes a este modelo para eliminar los problemas de actualización que presenta, como veremos en los próximos capítulos.

3.1.2.2. Dependencia Funcional (DF)

Dada una relación R, se dice que el atributo Y de R es funcionalmente dependiente del atributo X de R, si y sólo si, cada valor X en R tiene asociado a él, precisamente, un valor de Y en R en cualquier momento del tiempo.

Y depende funcionalmente de X se denota como:

$$X \rightarrow Y$$

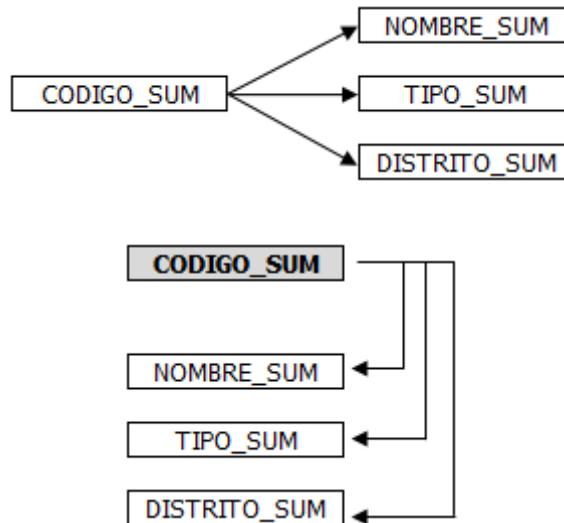
Observemos la siguiente tabla:

TABLA: SUMINISTRADOR

CÓDIGO_SUM	NOMBRE_SUM	TIPO_SUM	DISTRITO_SUM
S001	Maria Rojas Zamora	30	San Isidro
S002	Carla Segura Ramos	10	Miraflores
S003	Jenifer Lopez Arias	20	Lince
S004	Guadalupe Salvatierra Gonzales	20	Lince
S005	Janeth Cruz Cruz	15	Surco

Las columnas **NOMBRE_SUM**, **TIPO_SUM** y **DISTRITO_SUM** son funcionalmente dependientes cada uno de **CÓDIGO_SUM**, ya que para un valor de **CÓDIGO_SUM** existe un valor correspondiente de **NOMBRE_SUM**, **TIPO_SUM** y **DISTRITO_SUM**.

Estas cuatro (4) representaciones son formas de mostrar las dependencias funcionales.



$\text{CODIGO_SUM} \rightarrow \text{NOMBRE_SUM}$

$\text{CODIGO_SUM} \rightarrow \text{TIPO_SUM}$

$\text{CODIGO_SUM} \rightarrow \text{DISTRITO_SUM}$

$\text{CODIGO_SUM} \rightarrow \text{NOMBRE_SUM } \text{TIPO_SUM } \text{DISTRITO_SUM}$

El reconocimiento de las dependencias funcionales es una parte esencial de la comprensión de la semántica, del significado del dato. El hecho de que **DISTRITO_SUM** dependa funcionalmente de **CODIGO_SUM** significa que cada suministrador está situado en un distrito.

La noción de dependencia funcional puede ser extendida hasta cubrir el caso en que X, Y o ambos atributos sean compuestos.

3.1.2.3. Dependencia Funcional Completa (DFC)

El atributo Y es funcionalmente y completamente dependiente del atributo X, si es funcionalmente dependiente de X y no es funcionalmente dependiente de algún subconjunto de X.

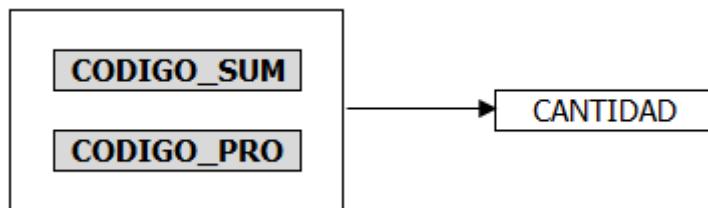
En algunos textos lo representan como: $X \implies\!> Y$

El atributo **CANTIDAD** es funcionalmente dependiente del par de atributos (**CODIGO_SUM** = Código del suministrador y **CODIGO_PRO** = Código del producto).

TABLA: SUM-PRO		
CÓDIGO_SUM	CÓDIGO_PRO	CANTIDAD
S001	P003	100
S002	P001	150
S002	P003	120
S005	P002	50
S004	P002	30
S001	P005	50
S003	P003	20
S001	P001	10
S002	P003	15
S005	P005	2
S001	P003	50
S002	P002	60
S001	P001	10
S004	P003	14

Por ello, se podría denotar **CANTIDAD** como **CANTIDAD_PRO_SUM**, que en el diccionario de datos se detallaría como la **cantidad de productos provista por un suministrador**.

Esta última explicación muestra que la cantidad tiene dependencia funcional completa tanto del suministrador como del producto (**Entidades Fuertes**).



3.1.2.4. Llave

Al hablar de entidad en el modelo entidad - relación, se asumió que existe una llave para cada entidad, formada por un conjunto de atributos que definen de forma única la entidad. Un concepto análogo se define para las relaciones o tablas en el modelo relacional.

Sean:

- R una relación con atributos A₁, A₂, ..., A_n y
- X un subconjunto de A₁, A₂, ..., A_n

Se dice que X es una llave de R si y solo si:

$$1) X \rightarrow A_1, A_2, \dots, A_n$$

Todos los atributos de la relación dependen funcionalmente de X

$$2) \exists Y \subset X \mid Y \rightarrow A_1, A_2, \dots, A_n$$

Lo planteado en el punto 2 es una condición de **minimalidad** que no se requería para el concepto de llave en el modelo entidad - relación.

Una relación puede tener varias llaves. Una de ellas se nombra "llave primaria" (la que se escoge para trabajar) y las restantes se nombran "**llaves candidatas**". Una **superllave** será cualquier superconjunto de una llave. Entonces, una llave es un caso especial de superllave.

3.1.2.5. Procedimiento para hallar las llaves candidatas de una relación

Supongamos que se quiere encontrar las llaves candidatas de una relación R (A, B, C, D, E) con las siguientes dependencias funcionales:

$$A \rightarrow B$$

$$BC \rightarrow D$$

$$AB \rightarrow E$$

Para comenzar, se parte de que no existen más llaves que dependencias funcionales, pues el concepto de llave incluye la existencia de dependencia funcional. Se analiza, por tanto, cada una de las **DF** presentes en la relación, añadiendo los atributos que sean

imprescindibles en la parte izquierda para lograr determinar a todos los atributos de la relación. El conjunto de atributos así formado debe ser mínimo.

Luego se analiza cada uno de esos conjuntos mínimos, de forma que, si alguno es un superconjunto de otro, ya no es llave, sino superllave. Pueden resultar varias llaves candidatas.

En el ejemplo:

1. A→B	AC→A B C D AC	es llave
2. BC→D	BCA→B C D A E BCA	es superllave
3. AB→E	ABC→A B E C D ABC	es superllave

La única llave es **AC**. No hay ninguna otra llave candidata, pues en las otras DF se obtiene el mismo conjunto ABC y este conjunto contiene a AC.

Ejemplo:

Sea la relación R (ciudad, calle, código postal). Para abreviar, R (C, A, P) donde se tiene que:

- Una calle en una ciudad tiene un código postal: CA→P.
- El código postal tiene una estructura tal que su valor determina la ciudad: P→C.
- Pero en una ciudad, varias calles pueden tener el mismo código, por
- lo que no se cumple P→A.

Entonces, analizando cada DF se tiene:

- CA→P, pero CA se determina a sí mismo también: CA→CA, por lo tanto, CA→CAP, es decir, que CA determina a todos los atributos de R y, como no existe ningún subconjunto de CA que, a su vez, determine a todos los atributos de la relación, se puede concluir que AC es llave.
- P → C y no determina a, pero agregando A en el lado izquierdo: PA→CA y es obvio que PA→PA también, entonces PA→CAP, y no existe ningún subconjunto de PA que, a su vez, determine a todos los atributos de la relación, por lo tanto, PA también es llave.

Atributo llave

Un atributo Ai de R es un atributo llave si él es (o es miembro de) una llave (candidata o primaria).

Atributo no llave

Un atributo Aj de R es un atributo no llave si él no es miembro de ninguna llave candidata.

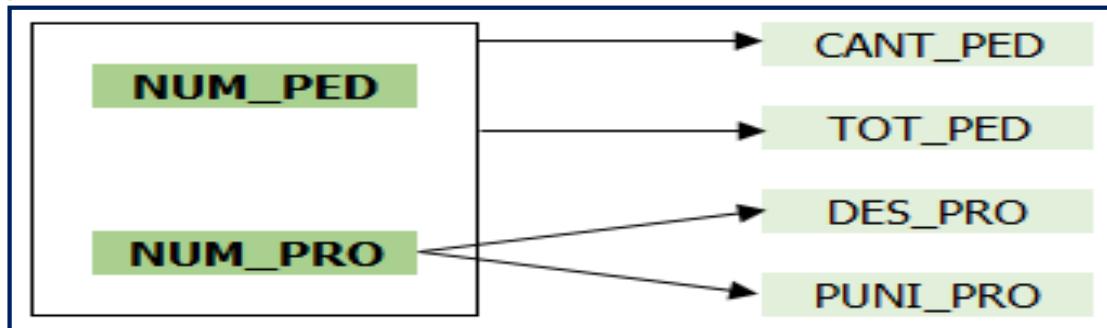
Por ejemplo, en el caso de la relación R (C, A, P) vista anteriormente, C, A y P son todos atributos llaves.

3.1.3. Segunda Forma Normal

Una relación R se dice que está en **2FN** si está en **1FN** y si, y solo si, los atributos no llaves (ni primarias, ni candidatas) de R, si los hubiese, son funcional y completamente dependientes de la llave primaria de R.

Entonces, se aplica solo a relaciones con llaves compuestas, pues no es posible que en una relación cuya llave primaria sea simple (compuesta por un solo atributo) haya atributos que dependan de parte de la llave primaria. Una relación que esté en 1FN y que tenga una llave primaria simple está en 2FN.

En el ejemplo de los pedidos de productos, habíamos visto que en la relación **PED-PRO** subsistían problemas de actualización. Analicemos las dependencias funcionales que existen en dicha relación:



Esta relación no está en 2FN, pues **DES_PRO** y **PUNI_PRO** no dependen funcional y completamente de la llave (**NUM_PED**, **NUM_PRO**).

Paso único. - Se determina si existen relaciones con clave compuesta. Si **NO** las hay, las relaciones obtenidas en la Primera Forma Normal se encuentran en Segunda Forma Normal. De lo contrario, se efectúa lo siguiente:

1. Se crea una relación para todos los atributos que dependen funcional y completamente de la llave (y los atributos que no se analizan por ser atributos llaves, pertenecientes a claves candidatas).

PED-PRO (NUM_PED, NUM_PRO, CANT_PED, TOT_PED)

2. Se crea una relación para los atributos que dependan de cada parte (subconjunto) de la llave. La llave de la relación así formada será la parte (subconjunto) de la llave primaria de la cual dependen los atributos.

PRODUCTO (NUM_PRO, DES_PRO, PUNI_PRO)

Los problemas planteados en la 1FN se resuelven con la 2FN. Veamos:

- **Creación.** Se puede insertar la información sobre un producto, aunque no haya un pedido que lo solicite.
- **Eliminación.** Se puede eliminar una línea de pedido y no se pierde la información sobre el producto, aunque sea el único pedido que pide ese producto.
- **Modificación.** Si cambia un atributo del producto, sólo hay que cambiarlo en un lugar. Se elimina redundancia.

Pero aún tenemos problemas en este caso que son similares a los vistos, pero con la relación **PEDIDO** y, específicamente, cuando se trata de insertar, eliminar o modificar la información de Clientes:

3.1.4. Tercera Forma Normal

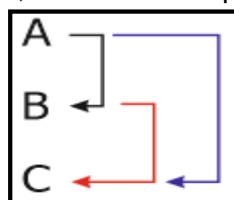
La tercera forma normal (3NF) es una forma normal usada en la normalización de bases de datos. La 3NF fue definida originalmente por E.F. Codd en 1971. La definición de Codd indica que una tabla está en 3NF si y solo si las dos condiciones siguientes se mantienen:

- La tabla está en la segunda forma normal (2NF)
- Ningún atributo no-primario de la tabla es dependiente transitivamente de una clave primaria

Esto es lo mismo que decir que se deben eliminar las dependencias transitivas de atributos no llaves respecto a la llave primaria, estando ya la relación en 2FN.

3.1.4.1. Dependencia transitiva

Sean **A**, **B** y **C** conjuntos de atributos de una relación R. Si B es dependiente funcionalmente de A y C lo es de B, entonces C depende transitivamente de A.



Este paso se ejecuta examinando todas las relaciones para ver si hay atributos no llaves que dependan unos de otros. Si se encuentran, se forma una nueva relación para ellos.

Veamos el siguiente caso: Desde una lista donde se visualiza el tipo de sorteo, el año en que se sorteó, el nombre del ganador y la fecha de nacimiento del ganador ya que solo pueden entrar al sorteo personas mayores de edad.

SORTEO	AÑO	GANADOR	FECHA DE NACIMIENTO
Diá de la Madre	2010	Maria Rosales Neyra	25/02/1980
Navidad	2010	Carlos Rojas Silva	05/08/1975
Diá de la Madre	2011	Guadalupe Pardo Ferrer	20/12/1982
Navidad	2011	Carlos Garcia Fernandez	30/04/1984
Diá de la Madre	2012	Fernanda Jimenez Gutierrez	04/05/1978
Navidad	2012	Gustavo Lucio Deza	14/11/1980

Si observamos la tabla el atributo ganador depende funcionalmente de sorteo y el año en que se realizó el sorteo, mientras que la fecha de nacimiento depende funcionalmente de ganador y transitivamente de sorteo y año.

Para solucionar esta anomalía separaremos las tablas en **Sorteo** y **Participantes** de la siguiente forma:

Tabla: Sorteo		
SORTEO	AÑO	GANADOR
Diá de la Madre	2010	Maria Rosales Neyra
Navidad	2010	Carlos Rojas Silva
Diá de la Madre	2011	Guadalupe Pardo Ferrer
Navidad	2011	Carlos Garcia Fernandez
Diá de la Madre	2012	Fernanda Jimenez Gutierrez
Navidad	2012	Gustavo Lucio Deza

Tabla: Participante

PARTICIPANTE	FECHA DE NACIMIENTO
Maria Rosales Neyra	25/02/1980
Carlos Rojas Silva	05/08/1975
Guadalupe Pardo Ferrer	20/12/1982
Carlos Garcia Fernandez	30/04/1984
Fernanda Jimenez Gutierrez	04/05/1978
Gustavo Lucio Deza	14/11/1980

Ahora las tablas se encuentran en 3FN separando el atributo transitivo a otra tabla, en donde este si dependa de la PK.

Veamos otro caso de 3FN; desde una lista donde se visualiza el número de DNI, nombre completo de un empleado, la especificación de su cargo y el monto de asignación salarial correspondiente según su cargo.

DNI	Empleado	Cargo	Salario
10584852	Jose Perez Roca	Jefe	S/. 3,500.00
20145175	Linda Torres Huaman	Contador	S/. 3,000.00
44578787	Carla Zegarra de la Cruz	Contador	S/. 3,000.00
36336955	Ivonne Sotomayor Rios	Administrativo	S/. 1,500.00

Como vemos el atributo “salario” no depende funcionalmente del número de DNI del empleado; entonces procedemos a separar el salario en una nueva entidad.

Cargo	Salario
Jefe	S/. 3,500.00
Contador	S/. 3,000.00
Administrativo	S/. 1,500.00

Finalmente, la entidad Empleado quedaría de la siguiente manera:

DNI	Empleado	Cargo
10584852	Jose Perez Roca	Jefe
20145175	Linda Torres Huaman	Contador
44578787	Carla Zegarra de la Cruz	Contador
36336955	Ivonne Sotomayor Rios	Administrativo

La regla de la **Tercera Forma Normal** establece que todas las dependencias parciales se deben eliminar y separar dentro de sus propias tablas. Una dependencia parcial es un término que describe a aquellos datos que no dependen de la llave primaria de la tabla para identificarlos.

Analicemos las dependencias funcionales que existen en la relación **PEDIDO**:

Paso 1: se remueven los atributos que no dependen de la llave

Se crea una relación para los atributos no llaves que no dependen transitivamente de la llave primaria (y los atributos que no se analizan por ser atributos llaves, pertenecientes a claves candidatas).

PEDIDO (NUM_PED, FEC_PED, NUM_CLI)

Paso 2: se remueven los atributos que dependen de la llave

Se crea una relación para los atributos no llaves que dependen transitivamente de la llave primaria a través de otro atributo o conjunto de atributos no llave primaria (que no son parte de la llave primaria.) La llave primaria de la relación así formada será el atributo o conjunto de atributos a través de los cuales existe la dependencia transitiva.

CLIENTE (NUM_CLI, NOM_CLI, DIR_CLI)

Es necesario analizar las otras relaciones. Puede comprobarse que en las otras relaciones no hay dependencia entre atributos no llaves, por lo que están en 3FN. Entonces, el modelo de datos relacional en **3FN** que representa el de los pedidos de productos está formado por las siguientes relaciones:

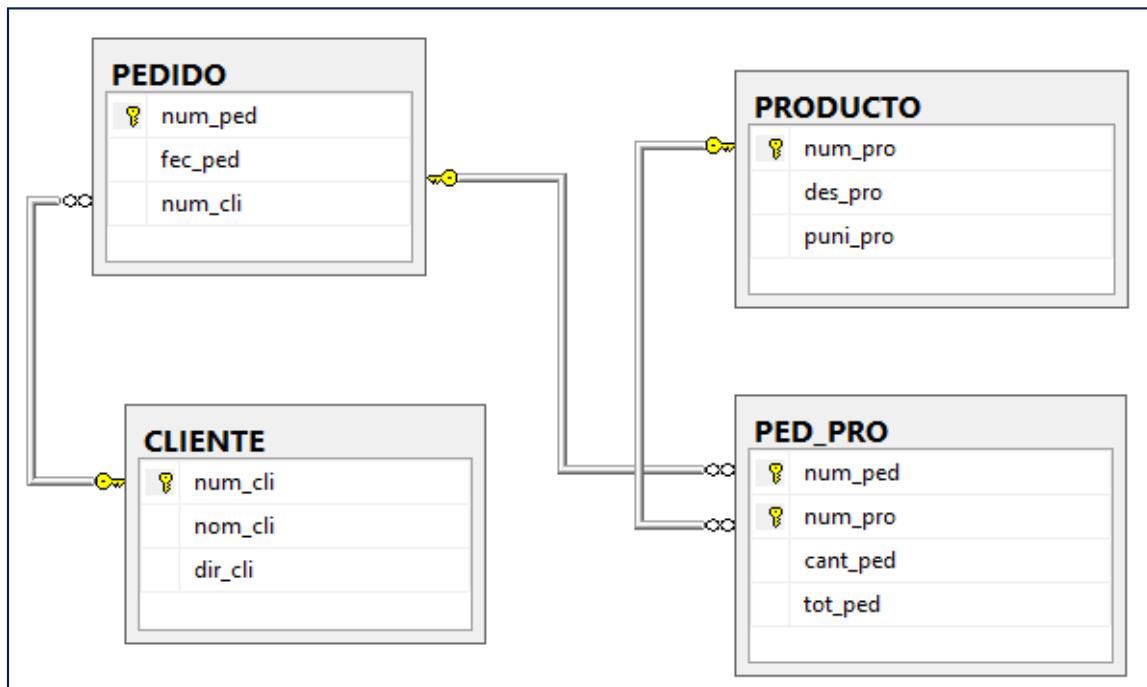
PEDIDO (NUM_PED, FEC_PED, NUM_CLI)

CLIENTE (NUM_CLI, NOM_CLI, DIR_CLI)

PED-PROD(Num_Ped, Num_Pro, CANT_PED, TOT_PED)

PRODUCTO(Num_Pro, DES_Pro, PUNI_Pro)

IMPLEMENTADO EN SQL SERVER 2014



SCRIPT

```
-- CREANDO LA BASE DE DATOS
CREATE DATABASE BD_COMERCIO
GO

--ACTIVANDO LA BASE DE DATOS
USE BD_COMERCIO
GO

--CREANDO LAS TABLAS
CREATE TABLE PRODUCTO
(num_pro char(7)      PRIMARY KEY,
des_pro varchar(70)   not null,
puni_pro money         not null
)
GO

CREATE TABLE CLIENTE
(num_cli int           PRIMARY KEY,
nom_cli varchar(80)   not null,
dir_cli varchar(60)   not null
)
GO
```

```
CREATE TABLE PEDIDO
(num_ped char(8)      PRIMARY KEY,
fec_ped date           not null,
num_cli int             not null,
foreign key (num_cli) REFERENCES CLIENTE
)
GO

CREATE TABLE PED_PRO
(num_ped char(8)      not null,
num_pro char(7)      not null,
cant_ped int          not null,
tot_ped money          not null,
primary key (num_ped,num_pro),
foreign key (num_ped) REFERENCES PEDIDO,
foreign key (num_pro) REFERENCES PRODUCTO
)
GO
```

La 3FN ha eliminado los problemas asociados con la información sobre el cliente en la 2FN. Veamos:

- ✓ **Creación.** - Se puede insertar la información de un cliente, aunque no haya un pedido para él.
- ✓ **Eliminación.** - Al borrar un pedido, que era el único que se le hacía a un cliente, no se perderá la información sobre el cliente.
- ✓ **Modificación.** - La información sobre un cliente está en una sola ocurrencia, por lo que, para cambiar cierta información de éste, sólo hay que hacerlo en dicha ocurrencia.

Finalmente, podemos decir que:

“Una tabla está normalizada en 3FN forma si todas las columnas que no son llave son funcionalmente dependientes por completo de la llave primaria y no hay dependencias transitivas. Una dependencia transitiva es aquella en la cual existen columnas que no son llave que dependen de otras columnas que tampoco son llave”.

3.1.5. Integración: Ejercicios propuestos

- Normalización I (Básicos)

Caso desarrollado 01: Aplique la primera forma normal al siguiente cuadro:

COD_ALU	NOM_ALU	APE_ALU	TELÉ_ALU
2015-0001	José Martin	Arias Sáenz	522-5626 / 985-852259
2015-0002	María Fernanda	López Cárdenas	986-589228 / 953-578578
2015-0003	Carla Rosario	Obregón Silva	415-6578
2015-0004	Juana Fiorella	Figueroa Rosas	985-157487 / 995-655447

Podemos observar que existen datos que no se encuentran atomizados como el número de teléfono. Entonces nuestra primera solución podría ser de la siguiente manera:

COD_ALU	NOM_ALU	APE_ALU	TELÉ_ALU
2015-0001	José Martin	Arias Sáenz	522-5626
2015-0001	José Martin	Arias Sáenz	985-852259
2015-0002	María Fernanda	López Cárdenas	986-589228
2015-0002	María Fernanda	López Cárdenas	953-578578
2015-0003	Carla Rosario	Obregón Silva	415-6578
2015-0004	Juana Fiorella	Figueroa Rosas	985-157487
2015-0004	Juana Fiorella	Figueroa Rosas	995-655447

Como podemos observar existen datos repetidos que deben ser controlados de la siguiente manera:

Alumno

<u>COD_ALU</u>	NOM_ALU	APE_ALU
2015-0001	José Martin	Arias Sáenz
2015-0002	María Fernanda	López Cárdenas
2015-0003	Carla Rosario	Obregón Silva
2015-0004	Juana Fiorella	Figueroa Rosas

Teléfono

<u>COD_ALU</u>	<u>TELÉ_ALU</u>
2015-0001	522-5626
2015-0001	985-852259
2015-0002	986-589228
2015-0002	953-578578
2015-0003	415-6578
2015-0004	985-157487
2015-0004	995-655447

Caso propuesto 01: Aplique la primera forma normal al siguiente cuadro:

COD_CLI	NOMBRES_CLI	TEL_CLI
2015-0001	José Martin Arias Sáenz	522-5626
2015-0002	María Fernanda López Cárdenas	986-589228
2015-0003	Carla Rosario Obregón Silva	415-6578
2015-0004	Juana Fiorella Figueroa Rosas	985-157487

Caso propuesto 02: Aplique la primera forma normal al siguiente cuadro:

DOCENTE	TIEMPO	CURSOS	ESCUELA	COORDINADOR
Juan Rojas R.	Completo	Introducción a la Algoritmia Estructura de Datos Base de Datos	Tecnología	SI NO SI
María Fernández C.	Parcial	Diseño Web I Diseño Web II Animación I	Diseño Gráfico	NO NO SI

Caso propuesto 02: Normalice el siguiente documento, según sea conveniente.

VILLANUEVA HERRERA DINO Mzp Lt12 Av. Tupac Inca Yupanqui Prov. Huarochiri TELÉFONO: 6160008	R.U.C. 10401381291 RECIBO POR HONORARIOS ELECTRONICO Nro: E001- 50
Recibí de: SOCIEDAD CENTROAMERICANA S.A.C	
Identificado con RUC	número 20145905678
Domiciliado en AV. PROCERES 4040 LIMA LIMA S.J.L	
La suma	CIENTO DIEZ Y 00/100 SOLES
Por concepto de REPARACION DE 1 EQUIPOS DE COMPUTO.	
Observación NINGUNA	
Inciso A DEL ARTÍCULO 33 DE LA LEY DEL IMPUESTO A LA RENTA	
Fecha de emisión 25 de Enero del 2017	
Total por honorarios: 110.00	
Retención (8 %) IR: (0.00)	
Total Neto Recibido: 110.00 SOLES	
No olvide registrar el pago de sus recibos por honorarios, para ello debe utilizar la opción "Registro de pagos", disponible en SUNAT Operaciones en Línea o a través del APP SUNAT: opción Recibo por Honorarios Electrónicos/	

3.1.6. Ejercicios de normalización básicos

- **Normalización I**

Para los siguientes casos determine los siguientes aspectos:

- Determine las dependencias funcionales suponiendo que inicialmente todos los datos se encuentran agrupados en una relación.
- Determine las llaves candidatas y señale la llave primaria.
- Diseñe el modelo de datos relacional normalizando las relaciones hasta la 3FN. Deben aparecer todas las relaciones que se vayan obteniendo en cada paso de normalización.

Caso propuesto 01: EMPRESA DE PROYECTOS



Se desea controlar la actividad de una empresa de proyectos. Para ello se cuenta con la siguiente información:

Del trabajador
Documento de identidad
Nombre completo
Salario

Del proyecto
Número de proyecto
Fecha de inicio
Fecha de finalización

Además, se conoce las horas en plan de trabajo que cada trabajador dedica a cada proyecto. Se sabe que un trabajador puede laborar en varios proyectos y en un proyecto participan varios trabajadores.

Finalmente, cada proyecto tiene una fecha de inicio-terminación y cada trabajador tiene un nombre y un salario, aunque un mismo salario o nombre puede serlo de varios trabajadores.

Caso propuesto 02: CONTROL DE VENTAS



En una tienda se desea controlar las ventas y se conoce:

- Código de cada mercancía, que la identifica.
- Descripción de cada mercancía.
- País de procedencia de cada mercancía.

- Moneda de cada país.
- Precio de venta de cada mercancía de una calidad dada.

Una mercancía tiene varias calidades (1, 2, 3, etc.) y una calidad puede serlo de varias mercancías, esta procede de un país y de un país proceden varias mercancías. Una mercancía tiene una descripción, pero una descripción puede serlo de mercancías con diferentes códigos.

Caso propuesto 03: CONTROL DE PUBLICACIONES



En una empresa editorial de libros se desea controlar las publicaciones que se realizan. Para ello, se tiene la siguiente información:

- De cada autor se conoce su código, que lo identifica, nombre y dirección.
 - Para cada libro se conoce su título, género literario y año en que fue escrito.
 - Un libro puede ser escrito por varios autores y un autor puede escribir varios libros.
 - Un libro puede tener varias ediciones y por cada edición de un libro, cada autor recibe un determinado pago.
 - De cada edición de un libro se conoce su fecha y cantidad de ejemplares.
- (Considere que los títulos de los libros no se repiten).

Caso propuesto 04: FACTURA DE VENTA

En el formulario que se presenta a continuación determine lo siguiente:

- a. Normalice el documento.
- b. Genere el modelo relacional correspondiente

R.T.N. 08019995298341			FECHA	FACTURA №	VENDEDOR
DIA	MES	ANO		00254	LENIN SALGADO
16	OCTUBRE	2007	NOMBRE DEL CLIENTE		
AUTO FRENOS BOULEVARD			TIPO DE FACTURA		
			CONTADO		
CANT.	CODIGO	DESCRIPCION	PRECIO UNITARIO	TOTAL	
10	7313570013	ADITIVO DE MOTOR 14.5 OZS	40.00	400.00	
3	731357012	ETER	60.00	180.00	
10	AD-35	BANDAS DENTADAS	2.50	25.00	
25	8-94218-497-0	RELAY PARA ISUZU 5P 12V	25.00	625.00	
75	40222-18000	TORNILLO DE RUEDA DATSUN 210	2.50	187.50	
10	90311-180754	RETENEDOR RUEDA DELANTERA TOYOTA	75.00	750.00	
2	4651	SILVINAS CAJA DE 12 UNIDADES	50.00	100.00	
30	1034AMARILLO	FOCOS DE STOP AMARILLO	7.20	216.00	
10	AT-111-35	FUSES HACHITA MINIATURA 35 AMP	1.80	18.00	
			SUB TOTAL	L. 2,501.50	
			I.S.V.	L. 300.18	
			TOTAL	L. 2,801.68	

1- No se aceptan cambios ni devoluciones
2- Despues de la fecha de vencimiento de esta factura devengara un recargo equivalente al interes maximo del sistema bancario
3- Por cada cheque devuelto por el banco se cobrarán Lps. 300.00 de recargo

"ORIGINAL"

Figura 21: Factura de venta

Fuente.- Tomado de http://www.sistematic21.com/factura_honduras.jpg

3.2. Tema 8 : Normalización Parte II

A continuación, vamos a profundizar y tratar los puntos mencionados en el tema 8 resolviendo casos y proponiendo.

Además, tener en cuenta lo siguiente:

- Cada formulario debe estar normalizado hasta la tercera forma normal (3FN)
- Debe generar el modelo relacional correspondiente para cada formulario.

3.2.1. Sin grupos repetitivos

Caso Desarrollado 01: CLUB CENTRO DEPORTIVO MUNICIPAL

CLUB CENTRO DEPORTIVO MUNICIPAL
FUNDADO EL 27 DE JULIO DE 1935

FICHA PERSONAL

SOCIO N°

DATOS PERSONALES

NOMBRES:

APELLIDOS:

LUGAR Y FECHA DE NACIMIENTO:

DOCUMENTO: DNI N°: OTRO:

DOMICILIO:

DISTRITO: PROVINCIA: DEPARTAMENTO:

TELEFONO CASA: CELULAR:

EMAIL:

ESTADO CIVIL: SOLTERO DIVORCIADO SEPARADO CASADO VIUDO OTRO

DATOS LABORALES

CENTRO DE TRABAJO:

DIRECCION:

TELEFONO:

EMAIL:

Calle Dominguez 436 San Borja, Lima - Perú Telefax: 346-1680
E-mail: deportivomunicipal@echamuni.net Web Site: <http://www.echamuni.net>

LISTADO DE ATRIBUTOS

SOCIO (NUMSOCIO, NOMBRE, APELLIDOS, FOTO, LUGARNAC, FECHANAC, NUMDNI, DOMICILIO, CODDISTRITO, NOMBREDISTRITO, CODPROVINCIA, NOMBREPROVINCIA, CODDEPARTAMENTO, NOMBREDEPARTAMENTO, TELEFCASA, TELEFMOVIL, EMAIL, CODESTADOCIVIL, DESCRIPCIONESTADOCIVIL, CODCENTRAB, NOMBRECENTRAB, DIRECCIONCENTRAB, TELEFCENTRAB, EMAILCENTRAB)

Se han agregado los códigos correspondientes a: Distrito, provincia, departamento, estado civil, centro de trabajo

APLICANDO FN1

No existen grupos repetitivos, por lo tanto, ya se encuentra en su 1FN.

SOCIO (NUMSOCIO, NOMBRE, APELLIDOS, FOTO, LUGARNAC, FECHANAC, NUMDNI, DOMICILIO, CODDISTRITO, NOMBREDISTRITO, CODPROVINCIA, NOMBREPROVINCIA, CODDEPARTAMENTO, NOMBREDEPARTAMENTO, TELEFCASA, TELEFMOVIL, EMAIL, CODESTADOCIVIL, DESCRIPCIÓNESTADOCIVIL, CODCENTRAB, NOMBRECENTRAB, DIRECCIÓNCENTRAB, TELEFCENTRAB, EMAILCENTRAB)

APLICANDO FN2

La clave primaria de la relación no es compuesta, por lo tanto, ya se encuentra en su 2FN.

SOCIO (NUMSOCIO, NOMBRE, APELLIDOS, FOTO, LUGARNAC, FECHANAC, NUMDNI, DOMICILIO, CODDISTRITO, NOMBREDISTRITO, CODPROVINCIA, NOMBREPROVINCIA, CODDEPARTAMENTO, NOMBREDEPARTAMENTO, TELEFCASA, TELEFMOVIL, EMAIL, CODESTADOCIVIL, DESCRIPCIÓNESTADOCIVIL, CODCENTRAB, NOMBRECENTRAB, DIRECCIÓNCENTRAB, TELEFCENTRAB, EMAILCENTRAB)

APLICANDO FN3

Analizamos dependencia transitiva.

SOCIO (NUMSOCIO, NOMBRE, APELLIDOS, FOTO, LUGARNAC, FECHANAC, NUMDNI, DOMICILIO, CODDISTRITO, TELEFCASA, TELEFMOVIL, EMAIL, CODESTADOCIVIL, CODCENTRAB)

ESTADOCIVIL (CODESTADOCIVIL, DESCRIPCIÓNESTADOCIVIL)

DISTRITO (CODDISTRITO, NOMBREDISTRITO, CODPROVINCIA)

PROVINCIA (CODPROVINCIA, NOMBREPROVINCIA, CODDEPARTAMENTO)

DEPARTAMENTO (CODDEPARTAMENTO, NOMBREDEPARTAMENTO)

CENTROTRABAJO (CODCENTRAB, NOMBRECENTRAB, DIRECCIÓNCENTRAB, TELEFCENTRAB, EMAILCENTRAB)

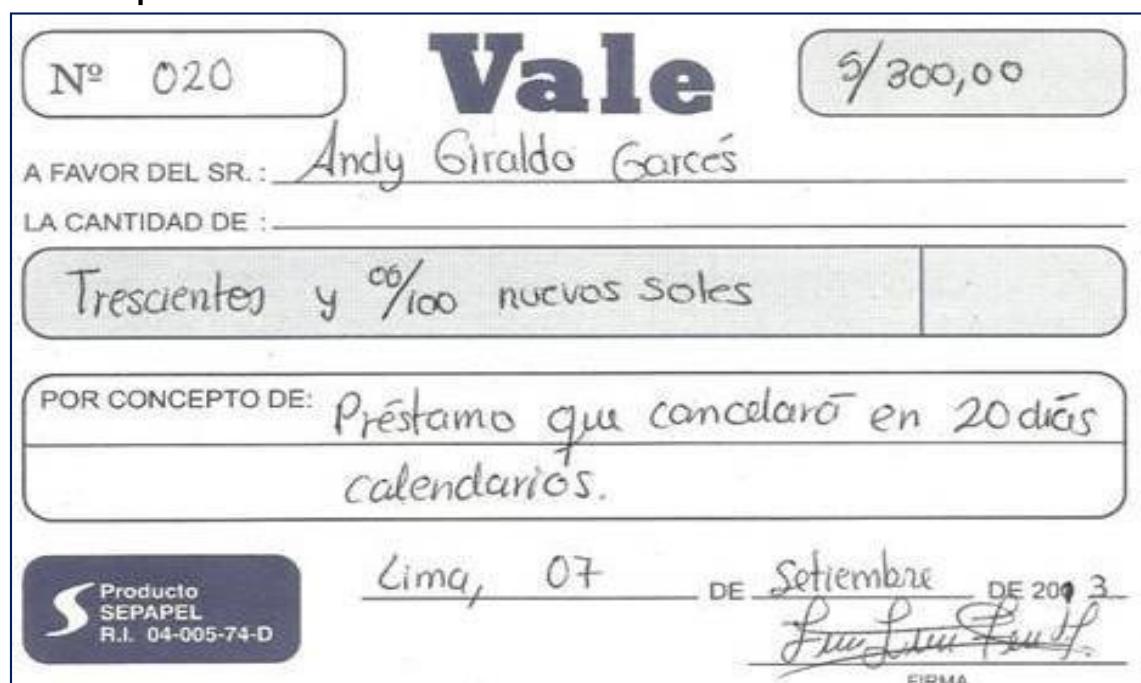
Caso Propuesto 01: VALE

Figura 22: Vale

Fuente.- Tomado de <http://www.monografias.com/trabajos96/documentosmercantiles/image021.jpg>

3.2.2. Con un grupo repetitivo

3.2.3. Ejercicios

Caso Desarrollado 01: ORDEN DE COMPRA

SOLUCIÓN:

ORDEN DE COMPRA

LISTADO DE DATOS ORIGINAL

ORDEN_COMPRA = (NRO_OC, NOM_PRV, FEC_PED, FEC_ENT, REQ_COM, NOM_SOLIC, FEC_PAG, TELF_SOLIC, CANT_ORD, NOM_PROD, PREC_UNI, IMPORTE, SUB_TOT, IVA, TOTAL, OBS)

LISTADO DE DATOS ACTUALIZADO

ORDEN_COMPRA = (NRO_OC, COD_PRV, NOM_PRV, FEC_PED, FEC_ENT, REQ_COM, COD_SOLIC, NOM_SOLIC, FEC_PAG, TELF_SOLIC, CANT_ORD, COD_PROD, NOM_PROD, PREC_UNI, NRO_OBS, DESCR_OBS)

LISTADO DE DATOS ACTUALIZADO Y LLAVE PRIMARIA

ORDEN_COMPRA = (NRO_OC, COD_PRV, NOM_PRV, FEC_PED, FEC_ENT, REQ_COM, COD_SOLIC,
NOM_SOLIC, FEC_PAG, TELF_SOLIC, CANT_ORD, COD_PROD, NOM_PROD, PREC_UNI, NRO_OBS,
DESCR_OBS)

IDENTIFICANDO ANOMALIA DE DATO NO ATÓMICO

ORDEN_COMPRA = (NRO_OC, COD_PRV, NOM_PRV, FEC_PED, FEC_ENT, REQ_COM, COD_SOLIC,
NOM_SOLIC, FEC_PAG, TELF_SOLIC, CANT_ORD, COD_PROD, NOM_PROD, PREC_UNI, NRO_OBS,
DESCR_OBS)

APLICANDO FN1

ORDEN_COMPRA = (**NRO_OC**, COD_PRV, NOM_PRV, FEC_PED, FEC_ENT, REQ_COM, COD_SOLIC, NOM_SOLIC, FEC_PAG, TELF_SOLIC)

OC_PRODUCTO = (NRO OC, COD PROD, NOM_PROD, PREC_UNI, CANT_ORD)

OC_OBSERVACIONES = (NRO OC, NRO OBS, DESCR_OBS)

IDENTIFICANDO ANOMALÍA DE DEPENDENCIA PARCIAL

ORDEN_COMPRA = (NRO OC, COD_PRV, NOM_PRV, FEC_PED, FEC_ENT, REQ_COM, COD_SOLIC, NOM_SOLIC, FEC_PAG, TELF_SOLIC)

OC_PRODUCTO (NRO OC, COD PROD, NOM PROD, PREC_UNI, CANT_ORD)

OC_OBSERVACIONES (NRO OC, NRO OBS, DESCR_OBS)

APLICANDO FN2

ORDEN_COMPRA = (NRO OC, COD_PRV, NOM_PRV, FEC_PED, FEC_ENT, REQ_COM, COD_SOLIC, NOM_SOLIC, FEC_PAG, TELF_SOLIC)

OC_PRODUCTO (NRO OC, COD PROD, PREC_UNI, CANT_ORD)

PRODUCTO = (COD PROD, NOM_PROD)

OC_OBSERVACIONES (NRO OC, NRO OBS, DESCR_OBS)

IDENTIFICANDO ANOMALÍA DE DEPENDENCIA TRANSITIVA

ORDEN_COMPRA = (NRO OC, COD_PRV, NOM PRV, FEC_PED, FEC_ENT, REQ_COM, COD_SOLIC, NOM SOLIC, FEC_PAG, TELF SOLIC)

OC_PRODUCTO (NRO OC, COD PROD, PREC_UNI, CANT_ORD)

PRODUCTO = (COD PROD, NOM_PROD)

OC_OBSERVACIONES (NRO OC, NRO OBS, DESCR_OBS)

APLICANDO FN3

ORDEN_COMPRA = (NRO OC, COD_PRV, FEC_PED, FEC_ENT, REQ_COM, COD_SOLIC, FEC_PAG)

PROVEEDOR = (COD PRV, NOM_PRV)

SOLICITANTE = (COD SOLIC, NOM_SOLIC, TELF_SOLIC)

OC_PRODUCTO = (NRO OC, COD PROD, PREC_UNI, CANT_ORD)

PRODUCTO = (COD PROD, NOM_PROD)

OC_OBSERVACIONES (NRO OC, NRO OBS, DESCR_OBS)

Caso Desarrollado 02: PROGRAMACIÓN DE VENDEDORES

El presente documento describe la asignación de vendedores a las distintas sucursales que tiene la empresa. En el documento se registran los datos de la sucursal y de los vendedores. Para normalizar correctamente se debe codificar los campos que tiene asterisco.

PROGRAMACIÓN DE VENDEDORES					
NÚMERO DE PROGRAMACIÓN:		FECHA DE PROGRAMACIÓN:		DISTRITO	UBICACIÓN DE SUCURSAL*:
SUCURSAL*		TIPO DE SUCURSAL* :		VENDEDORES	
VENDEDOR*	TURNO *	RUBRO DE VENTA*	ANEXO TELEFÓNICO	OBSERVACIÓN	

SOLUCIÓN

PROGRAMACIÓN DE VENDEDORES

LISTA DE ATRIBUTOS

PROGRAMA

NUMPROG, FECHAPROG, CODS, DESCRIPCIONES, CODTS, DESCRIPCIONTS, CODD, NOMBRED, CODV, NOMBREV, CODT, DESCRIPCIONT, CODRV, DESCRIPCIONRV, ANEXOTELEF, OBSERVACIÓN.

PRIMERA FORMA NORMAL – 1FN

Se identifican los grupos de atributos repetitivos.

PROGRAMA

NUMPROG, FECHAPROG, CODS, DESCRIPCIONES, CODTS, DESCRIPCIONTS, CODD, NOMBRED

PROGRAMA_VENDEDOR

NUMPROG, CODV, NOMBREV, CODT, DESCRIPCIONT, CODRV, DESCRIPCIONRV, ANEXOTELEF, OBSERVACIÓN.

VENDEDOR CODV,NOMBREV

SEGUNDA FORMA NORMAL – 2FN

Se analiza la dependencia funcional completa.

PROGRAMA

NUMPROG, FECHAPROG, CODS, DESCRIPCIONES, CODTS, DESCRIPCIONTS, CODD, NOMBRED

PROGRAMA_VENDEDOR

NUMPROG, CODV, CODT, DESCRIPCIONT, CODRV, DESCRIPCIONRV, ANEXOTELEF, OBSERVACIÓN.

TERCERA FORMA NORMAL – 3FN

Se analiza la dependencia transitiva.

PROGRAMA

NUMPROG, FECHAPROG, CODS

SUCURSAL

CODS, DESCRIPCIONES, CODTS

TIPO_SUCURSAL

CODTS, DESCRIPCIONTS, CODD

DISTRITO CODD, NOMBRED

PROGRAMA_VENDEDOR NUMPROG, CODV, CODT, CODRV, ANEXOTELEF, OBSERVACIÓN **TURNO**

CODT, DESCRIPCIONT

VENDEDOR CODV, NOMBREV

RUBROVENTA CODRV, DESCRIPCIONRV

Caso propuesto 01: FORMULARIO DE POSTULANTE

FORMULARIO DE POSTULANTE			
I. DATOS PERSONALES			
Nº POSTULANTE		NOMBRES Y APELLIDOS	
FECHA DE NAC.		SEXO: <input type="checkbox"/> MASCULINO <input type="checkbox"/> FEMENINO	
DIRECCION		DISTRITO:	
II. ESTUDIOS TECNICOS			
INSTITUTO	ESPECIALIDAD	CONCLUYO (SI/NO)	AÑO EN QUE CONCLUYO

Caso propuesto 02: GUÍA DE REMISIÓN

 IKAMI INDUSTRIAL S.A.C. Av. De los Andes N° 201 - Breña - Lima Av. César Vallejo N° 201 - Trujillo - La Libertad	RUC N° 21212121212 GUÍA DE REMISIÓN REMITENTE N° 001-0000041		
	Fecha de inicio del traslado: <u>12/07/2006</u>	Punto de partida: <u>Av. De los Andes N° 201, Breña, Lima</u>	
Destinatario: <u>Constructora Indurain E.I.R.L.</u>	Punto de llegada: <u>Av. América N° 2122, Trujillo, La Libertad</u>		
RUC: <u>20200876627</u> N° Doc. Identidad	Motivo del traslado:		
Venta <input checked="" type="checkbox"/> Venta sujeta a confirmación por el comprador <input type="checkbox"/> Compra <input type="checkbox"/> Traslado entre establecimientos de la misma empresa <input type="checkbox"/> Consignación <input type="checkbox"/> Devolución <input type="checkbox"/> Venta con entrega a terceros <input type="checkbox"/> Otros (especificar) _____			
Datos del bien transportado:			
Descripción: <u>Aqua de mesa x 20 lts. Marca "Pum"</u> Cantidad: <u>400</u> Unidad de medida: <u>Bidón</u> Peso: <u>8 TM</u>			
Datos del transportista:			
RUC: <u>20795365468</u>	Denominación, apellidos y nombres: <u>Carga Fácil S.A.C.</u>		
Datos de la Unidad de Transporte y conductor:			
Marca y placa	Licencia de conducir		
Imprenta Los Alceros S.R.L. RUC 21111154421 Fecha de impresión: 10.07.2006 N° de autorización: 321321321321			

3.2.4. Múltiples grupos repetitivos**3.2.5. Ejercicios****Caso propuesto 01: PRESUPUESTO DE OBRA**

El siguiente documento muestra un análisis de precios y requerimientos tanto de materiales, personal, así como de los equipos y herramientas necesarios para realizar una obra.

PRESUPUESTO DE OBRA					
Nº PRESUPUESTO			FECHA	/ /	
CLIENTE				DISTRITO	
DIRECCIÓN					
TELÉFONO					
MATERIALES					
DESCRIPCIÓN	UNIDAD	CANTIDAD	PRECIO UNITARIO	SUBTOTAL	
PERSONAL					
CARGO DEL PERSONAL	UNIDAD	CANTIDAD	COSTO UNITARIO	SUBTOTAL	
EQUIPOS Y HERRAMIENTAS					
DESCRIPCIÓN	UNIDAD	CANTIDAD	PRECIO UNITARIO	SUBTOTAL	
UNIDADES VALIDAS: U1: UNI - U2: ROLL - U3:GLB - U4: GLN - U5:DÍA - U6:HRS					TOTAL PRESUPUESTO

Figura 23: Presupuesto de obra
 Fuente. - Tomado desde la aplicación Excel

Caso propuesto 02: REGISTRO DE EVENTO

Se requiere registrar los diferentes speakers, personal, funciones, entre otros.

REGISTRO DE EVENTO				
NRO EVENTO				
TITULO EVENTO				
FECHA EVENTO				
LUGAR EVENTO		DIRECCIÓN	SALA	
SPEAKERS (expositores)				
NOMBRE	NACIONALIDAD		ESPECIALIDAD	
PERSONAL LOGISTICO				
DNI	NOMBRE	FUNCIÓN	HORA ING	HORA SAL

Figura 24: Registro de evento
 Fuente. - Tomado desde la aplicación Excel

Caso propuesto 03: PROGRAMACIÓN DE CAMPAÑA VISUAL

El presente documento se elabora cada vez que se desarrolla una campaña de salud, en este caso de tipo visual. En el documento se registran los datos del bus, el personal de trabajo y los materiales requeridos. Para normalizar correctamente se debe codificar los campos que tiene asterisco.

PROGRAMACIÓN DE CAMPAÑA VISUAL				
Nº Campaña:		Fecha		Tipo:*
I. DATOS DEL BUS Y DE LA ACTIVIDAD				
COD. BUS		ANTIGUEDAD		
CAPACIDAD DEL BUS		PAÍS DE PROCEDENCIA		
TIPO DE BUS*		MARCA		
II. DESCRIPCION DEL PERSONAL DE TRABAJO				
IdMédico	Nombre y Apellidos	Especialidad de Médico*	Horas de viaje acumuladas	Observación
III. MATERIALES REQUERIDOS				
Código	Descripción	Cantidad	Unidad medida*	

Figura 25: Programación de campaña visual

Fuente. - Tomado desde la aplicación Excel

Caso propuesto 04: FORMULARIO DE POSTULANTE

Se requiere registrar los diferentes institutos, especialidades, universidades, programas, entre otros.

FORMULARIO DE POSTULANTE			
I. DATOS PERSONALES			
Nº POSTULANTE		NOMBRES Y APELLIDOS	
FECHA DE NAC.		SEXO: <input type="checkbox"/> MASCULINO <input type="checkbox"/> FEMENINO	
DIRECCION		DISTRITO:	
II. ESTUDIOS TECNICOS			
INSTITUTO	ESPECIALIDAD	CONCLUYO (SI/NO)	AÑO EN QUE CONCLUYO
III. ESTUDIOS UNIVERSITARIOS			
UNIVERSIDAD	PROGRAMA	TITULO	AÑO EN QUE CONCLUYO

Figura 26: Formulario de postulante

Fuente. - Tomado desde la aplicación Excel

3.2.6. Diccionario de Datos

Un diccionario de datos es un conjunto de metadatos que contiene las características lógicas y puntuales de los datos que se van a utilizar en el sistema que se programa, incluyendo nombre, descripción, alias, contenido y organización.

- **Estructura de diccionario de datos.**

TABLA O ENTIDAD	ATRIBUTO O CAMPO	TIPO DE DATO	LONGITUD	LLAVE PRIMARIA	DESCRIPCIÓN	TABLA O ENTIDAD FORÁNEA
						A
Tabla 1	Atributo 1	Tipo A	XX	PK	Descripción del Atributo 1	
	Atributo 2	Tipo B	XX	FK	Descripción del Atributo 2	Tabla 2 (Atributo 1)
	Atributo 3	Tipo C	XX		Descripción del Atributo 3	
Tabla 2	Atributo 1	Tipo A	XX	PK	Descripción del Atributo 1	
	Atributo 2	Tipo B	XX		Descripción del Atributo 2	
	Atributo 3	Tipo C	XX		Descripción del Atributo 3	

- **Creación de diccionario de datos.**

TABLA O ENTIDAD	ATRIBUTO O CAMPO	TIPO DE DATO	LONGITUD	LLAVE PRIMARIA	DESCRIPCIÓN	TABLA O ENTIDAD FORÁNEA
Producto	Cod_Producto	Entero	5	PK	Código del Producto	
	Categoría	Entero	5	FK	Código de la categoría	Categorías(Cod_Categoría)
	Cod_Proveedor	Entero	5	FK	Código del proveedor	Proveedor(Cod_Proveedor)
	Nombre	Cadena	50		Nombre del producto	
	Precio	Decimal	-		Precio unitario del producto	
Categorías	Cod_Categoría	Entero	5	PK	Código de la categoría	
	Nombre	Cadena	50		Nombre de la Categoría	
Proveedores	Cod_Proveedor	Entero	5	PK	Código del Proveedor	
	Nombre	Cadena	50		Nombre del proveedor	
	Contacto	Cadena	100		Nombre de la persona de contacto	
	Fax	Numerico	10		Numero de Fax del proveedor	
Factura	N_Factura	Entero	5	PK	Numero correlativo de la factura	
	Cliente	Cadena	50		Nombre del Cliente	
	Fecha	DATE	-		Fecha de la Compra	
	Total	Decimal	-		Total de la Compra	
Detalle_Factura	N_Factura	Entero	5	FK	Número de la Factura	Factura(N_Factura)
	Cod_Producto	Entero	5	FK	Código del Producto	Producto(Cod_Producto)
	Cantidad	Entero	3		Cantidad de Producto	
	Precio	Decimal	-		Precio unitario del producto	
	Sub-total	Decimal	-		Subtotal de la cantidad por el precio	

3.3. Tema 9 : Normalización Parte III

A continuación, vamos a profundizar y tratar los puntos mencionados en el tema 9 resolviendo casos y proponiendo.

Además, tener en cuenta lo siguiente:

- Cada formulario debe estar normalizado hasta la tercera forma normal (3FN)
- Debe generar el modelo relacional correspondiente para cada formulario.

3.3.1. Normalización Detalle de Detalle / Ítems

Un paso importante dentro de la normalización de documentos, es la aplicación del detalle de ítems en la normalización, para eso veremos dos casos el primero aplicará las 3FN a un documento que presente detalle-ítem; mientras que el segundo aplicará las 3FN a un documento que presente detalle del detalle.

3.3.2. Ejercicio parte I

Caso Desarrollado 01: INFORME TÉCNICO (Detalle Ítems)

INFORME TECNICO					
NO					
CURSO					
SEMESTRE				SECCIÓN	
ESPECIALIDAD					
ITEM	PROBLEMAS				
1					
2					
3					
4					
ITEM	CAUSAS				
1					
2					
3					
4					
ITEM	EFEKTOS				
1					
2					
3					
4					
ITEM	ALTERNATIVAS				
1					
2					
3					
4					

Figura 27: Hoja de informe técnico
Fuente. - Tomado desde la aplicación Excel

Los números de ítems (problemas, causas, efectos y alternativas) se repiten en diferentes informes técnicos. Para dar solución al caso debemos seguir los siguientes pasos:

Pasos:

1. LISTANDO TODOS LOS ATRIBUTOS

```
INFORME ( NRO_INF, CURSO_INF, SEMESTRE_INF,
           SECCION_INF, ESPECIALIDAD_INF, FECHA_INF,
           NRO_ITEM_PRO, DESC_PRO,
           NRO_ITEM_CAU, DESC_CAU,
           NRO_ITEM_EFE, DESC_EFE,
           NRO_ITEM_ALT, DESC_ALT)
```

2. IDENTIFICANDO ANOMALÍA DE DATOS NO ATÓMICOS

```
INFORME ( NRO_INF, CURSO_INF, SEMESTRE_INF,
          SECCION_INF, ESPECIALIDAD_INF, FECHA_INF,
          NRO_ITEM_PRO, DESC_PRO,
          NRO_ITEM_CAU, DESC_CAU,
          NRO_ITEM_EFE, DESC_EFE,
          NRO_ITEM_ALT, DESC_ALT)
```

3. APLICANDO 1FN

```
INFORME ( NRO_INF, CURSO_INF, SEMESTRE_INF,
          SECCION_INF, ESPECIALIDAD_INF, FECHA_INF)
PRO-INF ( NRO_INF, NRO_ITEM_PRO, DESC_PRO)
CAU-INF ( NRO_INF, NRO_ITEM_CAU, DESC_CAU)
EFEC-INF ( NRO_INF, NRO_ITEM_EFE, DESC_EFE)
ALT-INF ( NRO_INF, NRO_ITEM_ALT, DESC_ALT)
```

4. APLICANDO 2FN

Identificando anomalía de dependencia transitiva:

```
INFORME ( NRO_INF,
          CURSO_INF,
          SECCION_INF, ESPECIALIDAD_INF,
          SEMESTRE_INF,
          FECHA_INF)
PRO-INF ( NRO_INF, NRO_ITEM_PRO, DESC_PRO)
CAU-INF ( NRO_INF, NRO_ITEM_CAU, DESC_CAU)
EFEC-INF ( NRO_INF, NRO_ITEM_EFE, DESC_EFE)
ALT-INF ( NRO_INF, NRO_ITEM_ALT, DESC_ALT)
```

5. APLICANDO 3FN

```
INFORME ( NRO_INF, SEMESTRE_INF, FECHA_INF,
          COD_CUR, COD_ESP, COD_SEC)

CURSO ( COD_CUR, DESCRIPCIÓN)

ESPECIALIDAD ( COD_ESP, DESCRIPCIÓN)

SECCION ( COD_SEC, DESCRIPCIÓN)

PRO-INF ( NRO_INF, NRO_ITEM_PRO, DESC_PRO)
```

PRO-INF (NRO_INF, NRO_ITEM_PRO, DESC_PRO)

CAU-INF (NRO_INF, NRO_ITEM_CAU, DESC_CAU)

EFEC-INF(NRO_INF, NRO_ITEM_EFE, DESC_EFE)

ALT-INF (NRO_INF, NRO_ITEM_ALT, DESC_ALT)

Caso Desarrollado 02: CONTROL DE ATENCIÓN (Detalle de Detalle)

Figura 28: Hoja de control de atención

Para dar solución al caso debemos seguir los siguientes pasos:

Pasos:

1 LISTANDO TODOS LOS ATRIBUTOS

CONTROL_ATENCION (NRO_CON, COD_ESP, DESC_ESP,
COD_DOC, NOM_DOC,
FECHA_CON,
COD_PAC, NOM_PAC,
DIA_PAC,
COD_MED, DESC_MED)

Donde:

Atributo	Descripción
NRO_CON	Número de control de atención por doctor.
COD_ESP	Código de la especialidad.
DES_ESP	Descripción de la especialidad del doctor.
FEC_CON	Fecha de registro del control de atención.

COD_PAC	Código del paciente.
NOM_PAC	Nombre completo del paciente.
DIA_PAC	Diagnóstico especificado por el doctor.
COD_MED	Código de medicamento.
DESC_ME	Descripción del medicamento especificado por el doctor según el diagnóstico realizado.

2. IDENTIFICANDO ANOMALÍAS DE DATOS NO ATÓMICO

**CONTROL_ATENCION (NRO_CON, COD_ESP, DESC_ESP,
COD_DOC, NOM_DOC,
FECHA_CON,**

)

COD_PAC, NOM_PAC,
DIA_PAC,

Grupos repetidos

COD_MED, DESC_MED

3. APlicando FN1

**CONTROL_ATENCION (NRO_CON, COD_ESP, DESC_ESP,
COD_DOC, NOM_DOC,
FECHA_CON)**

CONTROL_ATENCION_PACIENTE (NRO_CON, COD_PAC, NOM_PAC)

CONTROL_ATENCION_MEDICAMENTO (NRO_CON, COD_MED, DESC_MED)

4. IDENTIFICANDO ANOMALÍA DE DEPENDENCIA PARCIAL

**CONTROL_ATENCION (NRO_CON, COD_ESP, DESC_ESP,
COD_DOC, NOM_DOC,
FECHA_CON)**

CONTROL_ATENCION_PACIENTE (NRO_CON, COD_PAC, NOM_PAC)

CONTROL_ATENCION_MEDICAMENTO (NRO_CON, COD_MED, DESC_MED)

Esto se debe a que **NOM_PAC** solo depende de la llave **COD_PAC** y no de **NRO_CON**. Y que **DESC_MED** solo depende de **COD_MED** y no de **NRO_CON**.

5. APlicando FN2

**CONTROL_ATENCION (NRO_CON, COD_ESP, DESC_ESP,
COD_DOC, NOM_DOC,
FECHA_CON)**

CONTROL_ATENCION_PACIENTE (NRO_CON, COD_PAC)

PACIENTE (COD_PAC, NOM_PAC)

CONTROL_ATENCION_MEDICAMENTO (NRO_CON, COD_MED)

MEDICAMENTO (COD_MED, DESC_MED)

6. IDENTIFICANDO ANOMALÍA DE DEPENDENCIA TRANSITIVA

CONTROL_ATENCION (NRO_CON, COD_ESP,

DESC_ESP,

COD_DOC,

NOM_DOC,

FECHA_CON)

CONTROL_ATENCION_PACIENTE (NRO_CON, COD_PAC)

PACIENTE (COD_PAC, NOM_PAC)

CONTROL_ATENCION_MEDICAMENTO (NRO_CON, COD_MED)

MEDICAMENTO (COD_MED, DESC_MED)

Esto se debe a que el atributo **DESC_ESP** depende del atributo **COD_ESP** y no de la llave **NRO_CON**. Mientras que el atributo **NOM_DOC** depende del atributo **COD_DOC** y no de la llave principal **NRO_CON**.

7. APlicando FN3

CONTROL_ATENCION (NRO_CON, COD_ESP, COD_DOC, FECHA_CON)

ESPECIALIDAD (COD_ESP, DESC_ESP)

DOCTOR (COD_DOC, NOM_DOC)

CONTROL_ATENCION_PACIENTE (NRO_CON, COD_PAC)

PACIENTE (COD_PAC, NOM_PAC)

CONTROL_ATENCION_MEDICAMENTO (NRO_CON, COD_MED)

MEDICAMENTO (COD_MED, DESC_MED)

Caso Desarrollado 03: PARTE POLICIAL (Detalle de Detalle)

PARTE POLICIAL						
			NÚMERO: 2010123456 FECHA: 22/09/2016			
DATOS DEL AGENTE						
DOCUMENTO DE IDENTIDAD: 07252828				GRADO: TENIENTE		
NOMBRE: GERARDO				APELLO PATERNO: MACHUCA		
DATOS DE LOS DELINCUENTES						
Documento de Identificación	Nombre	Apellido Paterno	Dirección	Teléfono	Alias	Lugar de detención del delincuente en el Parte
07778899	FILISBERTO	MANCO	CALLE 1 111	460-06666	EL RICO BASURA	BARRACONES
077111222	ANA	FIEL	CALLE 1 112	-	CHICA DINAMITA	HUERTA PERDIDA
Cantidad de Delincuentes:						02
DETALLE DEL DELITO COMETIDO POR LOS DELINCUENTES EN EL PARTE						
Documento de Identificación del Delincuente	Código del Delito		Descripción del Delito			
07778899	DEL08		ROBO A MANO ARMADA			
07778899	DEL13		SECUESTRO			
077111222	DEL34		NO ESTUDIAR			
077111222	DEL08		ROBO A MANO ARMADA			

Figura 29: Parte Policial
 Fuente. - Tomado desde la aplicación Excel

SOLUCIÓN:**PARTE POLICIAL****LISTADO DE DATOS ORIGINAL Y ACTUALIZADO**

PARTE POLICIAL (NROPARTE, FECHA, CODGRADO, DESCRIPGRADO, NRO_DOC_AGEN, NOMBREAGEN, APEPATERAGEN, APEMATERAGEN, NRO_DOC_DELIN, NOMBREDELIN, APEPATERDELIN, DIRECCDELIN, TLFDELIN, ALIASDELIN, LUGARDETDELIN, NRO_DOC_DELIN, CODDELITO, DESCRIPDELITO)

LLAVE PRIMARIA

PARTE POLICIAL (NROPARTE, FECHA, CODGRADO, DESCRIPGRADO, NRO_DOC_AGEN, NOMBREAGEN, APEPATERAGEN, APEMATERAGEN, NRO_DOC_DELIN, NOMBREDELIN, APEPATERDELIN, DIRECCDELIN, TLFDELIN, ALIASDELIN, LUGARDETDELIN, NRO_DOC_DELIN, CODDELITO, DESCRIPDELITO)

IDENTIFICANDO ANOMALIA DE DATO NO ATÓMICO

PARTE POLICIAL (NROPARTE, FECHA, CODGRADO, DESCRIPGRADO, NRO_DOC_AGEN, NOMBREAGEN, APEPATERAGEN, APEMATERAGEN,

NRO_DOC_DELIN, NOMBREDELIN, APEPATERDELIN, DIRECCDELIN, TLFDELIN, ALIASDELIN, LUGARDETDELIN,

NRO_DOC_DELIN, CODDELITO, DESCRIPDELITO)

APLICANDO FN1

PARTE POLICIAL (NROPARTE, FECHA, **CODGRADO, DESCRIPGRADO, **NRO_DOC_AGEN**, NOMBREAGEN, APEPATERAGEN, APEMATERAGEN)**

PARTE_DELINCUENTE (NROPARTE, **NRO_DOC_DELIN, NOMBREDELIN, APEPATERDELIN, DIRECCDELIN, TLFDELIN, ALIASDELIN, LUGARDETDELIN)**

PARTE_DELINCUENTE_DELITO (NROPARTE, **NRO_DOC_DELIN, **CODDELITO**, DESCRIPDELITO)**

IDENTIFICANDO ANOMALÍA DE DEPENDENCIA PARCIAL

PARTE POLICIAL (NROPARTE, FECHA, **CODGRADO, DESCRIPGRADO, **NRO_DOC_AGEN**, NOMBREAGEN, APEPATERAGEN, APEMATERAGEN)**

PARTE_DELINCUENTE (NROPARTE, **NRO_DOC_DELIN, NOMBREDELIN, APEPATERDELIN, DIRECCDELIN, TLFDELIN, ALIASDELIN, LUGARDETDELIN)**

PARTE_DELINCUENTE_DELITO (NROPARTE, **NRO_DOC_DELIN, **CODDELITO**, DESCRIPDELITO)**

APLICANDO FN2

PARTE POLICIAL (NROPARTE, FECHA, **CODGRADO, DESCRIPGRADO, **NRO_DOC_AGEN**, NOMBREAGEN, APEPATERAGEN, APEMATERAGEN)**

PARTE_DELINCUENTE (NROPARTE, **NRO_DOC_DELIN, LUGARDETDELIN)
DELINCUENTE (NRO_DOC_DELIN, NOMBREDELIN, APEPATERDELIN, DIRECCDELIN, TLFDELIN, ALIASDELIN)**

PARTE_DELINCUENTE_DELITO (NROPARTE, **NRO_DOC_DELIN, **CODDELITO**)**

DELITO (CODDELITO, DESCRIPDELITO)

IDENTIFICANDO ANOMALÍA DE DEPENDENCIA TRANSITIVA

PARTE POLICIAL (NROPARTE, FECHA, **CODGRADO, DESCRIPGRADO, **NRO_DOC_AGEN**, NOMBREAGEN, APEPATERAGEN, APEMATERAGEN)**

PARTE_DELINCUENTE (NROPARTE, **NRO_DOC_DELIN, LUGARDETDELIN)**

DELINCUENTE (NRO_DOC_DELIN, NOMBREDELIN, APEPATERDELIN, DIRECCDELIN, TLFDELIN, ALIASDELIN)

PARTE_DELINCUENTE_DELITO (NROPARTE, **NRO_DOC_DELIN, **CODDELITO**)**

DELITO (CODDELITO, DESCRIPDELITO)

APLICANDO FN3

PARTE POLICIAL (NROPARTE, FECHA, **NRO_DOC_AGEN)**

AGENTE (NRO_DOC_AGEN, NOMBREAGEN, APEPATERAGEN, APEMATERAGEN, **CODGRADO)**

GRADO (CODGRADO, DESCRIPGRADO)

PARTES DELINCUENTES (NRO PARTE, NRO DOC DELIN, LUGAR DET DELIN)

DELINCUENTE (NRO_DOC_DELIN), NOMBREDELIN, APEPATERDELIN, DIRECCDELIN,
TLFDELIN, ALIASDELIN)

PARTE_DELINCUENTE_DELITO (NROPARTE, NRO_DOC_DELIN, CODDELITO)

DELITO (CODDELITO, DESCRIPDELITO)

3.3.3. Ejercicios Propuestos: Ítems Detalle / Detalle de Detalle

Tener en cuenta lo siguiente:

- Cada formulario debe estar normalizado hasta la tercera forma normal (3FN)
 - Debe generar el modelo relacional correspondiente para cada formulario.

Caso propuesto 01: CONTROL DE PRODUCCIÓN

El siguiente documento muestra el control de prendas que realiza un determinado empleado, así como el control del tiempo y las ocurrencias por semana.

Figura 30: Hoja de control de producción

Caso Propuesto 02: INFORME DE SERVICIO (ítems)

INFORME DE SERVICIO									
	N° _____								
FECHA _____	ASUNTO _____								
TECNICO _____	DIRECCION _____								
CLIENTE _____	_____								
TELEFONO _____	_____								
<table border="1"> <thead> <tr> <th>ITEM</th> <th>OBSERVACIONES</th> </tr> </thead> <tbody> <tr> <td>1</td> <td> </td> </tr> <tr> <td>2</td> <td> </td> </tr> <tr> <td>3</td> <td> </td> </tr> </tbody> </table>		ITEM	OBSERVACIONES	1		2		3	
ITEM	OBSERVACIONES								
1									
2									
3									
<table border="1"> <thead> <tr> <th>ITEM</th> <th>CONCLUSIONES</th> </tr> </thead> <tbody> <tr> <td>1</td> <td> </td> </tr> <tr> <td>2</td> <td> </td> </tr> <tr> <td>3</td> <td> </td> </tr> </tbody> </table>		ITEM	CONCLUSIONES	1		2		3	
ITEM	CONCLUSIONES								
1									
2									
3									
<table border="1"> <thead> <tr> <th>ITEM</th> <th>RECOMENDACIONES</th> </tr> </thead> <tbody> <tr> <td>1</td> <td> </td> </tr> <tr> <td>2</td> <td> </td> </tr> <tr> <td>3</td> <td> </td> </tr> </tbody> </table>		ITEM	RECOMENDACIONES	1		2		3	
ITEM	RECOMENDACIONES								
1									
2									
3									

Caso Especial

Caso Propuesto 03: CONTRATO DE SERVICIOS (Detalle de Detalle)

1. Datos Generales				
Vendedor	Fecha Inicio			
Cliente	Meses de Contrato			
2. Servicio Contratado				
Servicio	Especificación	Precio	Beneficios	
			código	descripción
Telefonía Fija				
Cable				
...	
			...	
			...	
			...	

Caso Propuesto 04: HOJA DE PARTIDO DE FÚTBOL PROFESIONAL

HOJA DE PARTIDO DE FUTBOL PROFESIONAL				
	nº partido			
Campeonato				
Fecha Campeonato				
Fecha			Hora	
Estadio				
ÁRBITROS				
Nombres	fecha nac	dni		
			PRINCIPAL	
			LINEMAN IZQ.	
			LINEMAN DER	
			CUARTO OFICIAL	
EQUIPO L				
COMANDO TECNICO				
Nombre	Función			
	DT			
	ASISTENTE			
	MEDICO			
JUGADORES				
nro	nombre	posición		
1				
2				
3				
4				
5				
6				
EQUIPO V				
COMANDO TECNICO				
Nombre	Función			
	DT			
	ASISTENTE			
	MEDICO			
JUGADORES				
nro	nombre	posición		
1				
2				
3				
4				
5				
6				

Caso Propuesto 05: HOJA DE PARTIDO DE FÚTBOL PROFESIONAL

REGISTRO DE CONTROL DE JUGADORES					
			NÚMERO : 1020304512		
			FECHA : 09/05/2016		
DATOS DEL REPRESENTANTE					
DNI_REPRESENTANTE		40568754		APELLIDOS	GUERRERO RIVAS
NOMBRE		JUAN JOSÉ		PAÍS_REPRESENTANTE	ECUADOR
DATOS DE LOS JUGADORES					
40138130	JOSÉ	QUISPE VELA	25	990990365	PULPIN
40138125	ANDRÉS	LAGOS TORRES	22	987878797	EL RAYO
40138155	PEDRO	VÉLIZ HERRERA	26	989898898	EL PULPO
				Cantidad de Jugadores	3
DETALLE DE CLUBES QUE PERTENECIERON LOS JUGADORES					
40138130	E0001	ALIANZA LIMA	2 AÑOS Y 6 MESES	PRIMERA	
40138130	E0005	CRISTAL	1 AÑO	PRIMERA	
40138125	E0008	ALIANZA LIMA	3 AÑOS	PRIMERA	
40138155	E0010	UNIVERSITARIO	1 AÑO Y 6 MESES	PRIMERA	
40138155	E0013	CIENCIANO	1 AÑO	SEGUNDA	
40138155	E0012	SAN MARTÍN	2 AÑOS Y 6 MESES	PRIMERA	

Figura 31: Registro de control de jugadores

Fuente. - Tomado desde de la aplicación Excel

Caso Propuesto 06: MANTENIMIENTO DE VEHÍCULOS

El siguiente documento registra a un vehículo para su mantenimiento y/o reparación. Al ingresar un vehículo se hace un inventario de los accesorios con los que cuenta, luego el mecánico que atiende registra el diagnóstico y/o reparación que realiza. Por último, en el presente documento se registran las ocurrencias indicando el turno y fecha.

MANTENIMIENTO DE VEHÍCULOS					
NRO. MANTENIMIENTO			FECHA REGISTRO		
EMPLEADO			PLACA		
MARCA			MODELO		
HORA DE ENTRADA	FECHA SALIDA		HORA SALIDA		
INVENTARIO					
CÓDIGO ACCESORIO	ACCESORIO		ESTADO		
DIAGNÓSTICO/ REPARACIÓN					
CÓDIGO CÓDIGO	DIAGNÓSTICO/ REPARACIÓN	MECANICO MECÁNICO			
* Anotar todas las ocurrencias					
OCURRENCIAS					
ÍTEM	OCURRENCIA	TURNO	FECHA		
TURNO : M: MAÑANA T : TARDE N: NOCHE					

Figura 32: Mantenimiento de vehículos
Fuente. - Tomado desde de la aplicación Excel

3.3.4. Ejercicios Propuestos (Adicionales)

En los siguientes ejercicios que se presentan tener en cuenta lo siguiente:

- Determine las dependencias funcionales suponiendo que inicialmente todos los datos se encuentran agrupados en una relación.
- Determine las llaves candidatas y señale la llave primaria.
- Diseñe el modelo de datos relacional normalizando las relaciones hasta la 3FN. Deben aparecer todas las relaciones que se vayan obteniendo en cada paso de normalización.

Ejercicio #1

En una videoteca se desea diseñar una base de datos para el control de los préstamos de películas. De cada película esta se conoce su código, que la identifica, su título y la duración que tiene. Una película es de un género y un género puede serlo de muchas películas. De cada género se conoce su código, que lo identifica y su nombre. De cada película se tienen muchas copias, pero una copia lo es de una sola película. De cada copia se sabe su código, que la identifica, la marca del casete en que está grabada y su

estado de conservación. Una copia se les puede prestar a muchos usuarios y a un usuario se le pueden prestar muchas copias. No obstante, se sabe que a un determinado usuario, en una fecha dada, se le presta una sola copia. También se sabe que una determinada copia, en una fecha dada, se le presta a un solo usuario. De cada usuario se conoce su DNI, que lo identifica, su nombre, su dirección y su ocupación.

Ejercicio #2

En una empresa de productos de acero se desea controlar la producción total mensual y anual, y la producción mensual por talleres de cada producto, para lo cual cuenta con la siguiente información:

- Código de cada producto, que lo identifica.
- Descripción de cada producto.
- Peso de cada producto.
- Producción mensual total de cada producto.
- Producción mensual por taller de cada producto.
- Número de cada taller, que lo identifica.
- Se sabe que un producto puede ser producido por diferentes talleres y en diferentes meses, y un taller produce cada mes varios productos, aunque no todos los meses se producen los mismos productos en los mismos talleres.
- Considere que la información se almacena para un año.

Ejercicio #3

En un Hospital de Lima se desea controlar la atención que se brinda a los pacientes en consultas externas. Un paciente puede requerir atención médica en varias especialidades y en una especialidad se atienden muchos pacientes. De cada paciente se conoce su DNI, su nombre, (que puede repetirse en diferentes pacientes), su edad y su sexo. De cada especialidad se tiene un código, que la identifica, su descripción y la cantidad de médicos de esa especialidad que laboran en consultas externas. Un médico atiende sólo una especialidad, aunque en una especialidad laboran muchos médicos. Él atiende a muchos pacientes y un paciente es atendido por varios médicos. De cada médico se sabe su DNI, su nombre, (que puede coincidir con el de otro médico), el año de su graduación y el grado que tiene como médico (especialista de 1er grado, 2do grado, etc.). Se conoce para un paciente, atendido por un médico, el diagnóstico y el tratamiento indicado.

Ejercicio #4

En una biblioteca se desea diseñar una base de datos para el control de los préstamos de libros. De cada libro se conoce el código que lo identifica, su título y la cantidad de páginas que tiene. Un libro se clasifica por una materia y por una materia se clasifican muchos libros. De cada materia se conoce el código que la identifica y su nombre. De cada libro se tienen muchos ejemplares, pero un ejemplar lo es de un solo libro. De cada ejemplar se sabe el código que lo identifica, y su estado de conservación. Un ejemplar se les puede prestar a muchos usuarios y a un usuario se le pueden prestar muchos ejemplares. No obstante, se sabe que a un determinado usuario, en una fecha dada, se le presta un solo ejemplar. También se sabe que un determinado ejemplar, en una fecha dada, se le presta a un solo usuario. De cada usuario se conoce su DNI, su nombre, su dirección y su ocupación.

Ejercicio #5

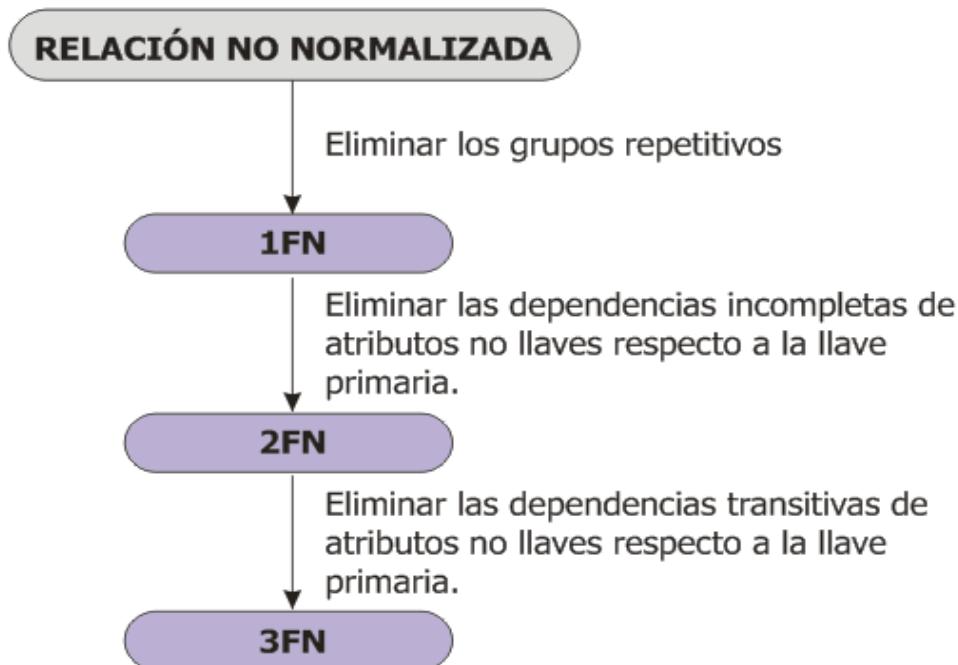
En un taller de mantenimiento de equipos de computación se desea diseñar una base de datos para el control de los servicios brindados. En el taller laboran varios trabajadores. De cada uno se conoce su DNI, su nombre y categoría laboral. De cada tipo de mantenimiento se sabe el código que lo identifica, su descripción y la frecuencia con que debe realizarse. Un trabajador puede realizar distintos tipos de mantenimiento y un tipo de mantenimiento puede ser ejecutado por diversos trabajadores. El mantenimiento de los equipos de computación se ejecuta por órdenes de servicio. De cada orden de servicio se conoce el número que la identifica, la empresa que la solicita y la fecha en que se solicita. Una orden de servicio puede contener la solicitud de mantenimiento de varios tipos de equipos y puede solicitarse el mantenimiento de un tipo de equipo en diferentes órdenes de servicio. De cada tipo de equipo se sabe el código que lo identifica, su descripción y fabricante. A cada tipo de equipo se le pueden realizar diferentes tipos de mantenimiento y un mismo tipo de mantenimiento se le puede realizar a diferentes tipos de equipos, pero se sabe que, en una orden de servicio dada, para un tipo de equipo, sólo se solicita ejecutar un único tipo de mantenimiento y también se conoce que un solo trabajador lo ejecuta. Además, en una orden de servicio dada, para un tipo de equipo, se sabe la cantidad de equipos que se solicita mantener. En una orden de servicio se pueden solicitar distintos tipos de mantenimiento y un tipo de mantenimiento se puede solicitar en distintas órdenes de servicio, pero está establecido que, en una orden de servicio, un tipo de mantenimiento dado, sólo se solicita para un único tipo de equipo.

Ejercicio #6

Se desea diseñar una base de datos para controlar la disponibilidad de materiales de construcción. De cada proveedor de materiales se conoce el código (**cprov**) que lo identifica, su nombre (**nomprov**) y el distrito en que radica (**dist**). De cada material se sabe el código (**cmat**) que lo identifica, su descripción (**desc**), la unidad de medida que se aplica al material (**um**) y el precio por unidad de medida (**precio**). Para guardar estos materiales hasta su posterior distribución, existen diversos almacenes. De cada almacén se conoce el código (**calm**) que lo identifica, su dirección (**diralm**) y la capacidad de almacenaje (**capac**). Un proveedor puede suministrar varios materiales y un material puede ser suministrado por diferentes proveedores. Se sabe que un material suministrado por un proveedor está en un solo almacén y, además, se sabe qué cantidad de un material suministrado por un proveedor se encuentra en el almacén (**cantmat**). En un almacén se guardan distintos materiales y pueden existir varios almacenes donde se guarde un mismo material.

Resumen

1. En el modelo relacional, tanto las entidades como las relaciones se representan como relaciones (tablas). Las ocurrencias (de las entidades o de las relaciones) se almacenan como filas y las columnas son los atributos (de las entidades o de las relaciones).
2. Para toda tabla del modelo relacional se cumple lo siguiente:
 - a) No hay dos filas (tuplas) iguales.
 - b) El orden de las filas no es significativo.
 - c) El orden de las columnas no es significativo.
 - d) Cada valor de un atributo es un dato atómico (o elemental).
3. La normalización es la expresión formal del modo de realizar un buen diseño. Con la aplicación de la teoría de la normalización se evitan los problemas de actualización.
4. La normalización está constituida por pasos sucesivos:
 - a) Primera Forma Normal (1FN)
 - b) Segunda Forma Normal (2FN)
 - c) Tercera Forma Normal (3FN)
 - d) Forma Normal de Boyce-Codd (FNBC)
“Para que una relación esté en 2FN tiene que estar en 1FN, para que esté en 3FN tiene que estar en 2FN y así sucesivamente”.
5. Una relación está en 1FN si no incluye ningún grupo repetitivo.
6. Para transformar una relación que no está en 1FN en otras que sí lo estén y representen mejor el, se deben eliminar los grupos repetitivos, para lo cual:
 - a) Se dejan en la relación original sólo los atributos que no son repetitivos.
 - b) Se extraen en una nueva relación los grupos repetitivos, escogiendo adecuadamente la llave primaria, que siempre estará formada por la llave de la relación original más uno o varios de los atributos repetitivos que permitan distinguir cada valor diferente en dichos campos.
7. Dada una relación R, se dice que el atributo Y de R es funcionalmente dependiente del atributo X de R, si y sólo si cada valor X en R tiene asociado a él, precisamente, un valor de Y en R en cualquier momento del tiempo.
8. El atributo Y es funcionalmente y completamente dependiente del atributo X, si es funcionalmente dependiente de X y no es funcionalmente dependiente de algún subconjunto de X.
9. Una relación R se dice que está en 2FN si está en 1FN y si, y sólo si, los atributos no llaves (ni primarias, ni candidatas) de R, si los hubiese, son funcional y completamente dependientes de la llave primaria de R.
10. Una relación R está en 3FN si está en 2FN y si, y sólo si, los atributos no llaves son independientes de cualquier otro atributo no llave primaria. Esto es lo mismo que decir que se deben eliminar las dependencias transitivas de atributos no llaves respecto a la llave primaria, estando ya la relación en 2FN.
11. Un determinante es cualquier atributo o conjunto de atributos del cual depende funcional y completamente cualquier otro atributo. Mejor dicho, la parte izquierda de la implicación cuando la dependencia funcional es completa.
12. De modo resumido, se puede decir que los pasos que se deben dar en el proceso de normalización son:



Si desea saber más acerca de estos temas, puede consultar las siguientes páginas:

Link	Descripción
http://dataoteca.unad.edu.co/contenidos/301330/Contenido_Linea_EXE-1/capitulo_6_normalizacin.html	En esta página web se muestra conceptos complementarios de cada paso de la normalización de datos.
http://www.mitecnologico.com/Main/ModeloErYNormalizacion	Es esta página web encontrará definiciones complementarias a la normalización de datos.
https://nelwibaez.files.wordpress.com/2011/04/guc3ada-de-ejercicios_iii_normalizac3b3n2.docx	Es un archivo de Word que contiene un ejemplo completo de normalización; así como propuestos.
http://users.dsic.upv.es/asignaturas/afe/inf/es/solucion_BD.pdf	Es esta página web encontrará casos desarrollados en Modelo Entidad Relación.
https://sites.google.com/site/modelami/entodebasesdedatos/clase-4	Es esta página web encontrará los conceptos básicos de normalización; así como un caso desarrollado paso a paso hasta la tercera forma normal.

4

DIAGRAMA ENTIDAD RELACIÓN

LOGRO DE LA UNIDAD DE APRENDIZAJE

Al término de la cuarta unidad, el estudiante obtiene el modelo lógico (DER a MER); así como el modelo físico (MER a DER), del resultante de organizar datos no estructurados.

TEMARIO

4.1. Tema 10 : **Modelo lógico (DER a MER)**

4.1.1 : Obtención del modelo lógico-global de los datos a partir del DER

4.1.2 : Modelo lógico. Ejercicios

4.2. Tema 11 : **Modelo físico (MER a DER)**

4.2.1 : Metodología de sistema de base de datos

4.2.2 : Obtención del modelo lógico-global de los datos a partir del MER

ACTIVIDADES PROPUESTAS

1. Explicaciones por parte del facilitador combinadas con ejemplos y preguntas de comprobación dirigidas a los alumnos.
2. Se realizan ejercicios por parte de los alumnos con la orientación del facilitador.

4.1. TEMA 10: MODELO LÓGICO (DER A MER)

El modelo lógico traduce los escenarios de uso creados en el diseño conceptual en un conjunto de objetos de negocio y sus servicios. El modelo lógico se convierte en parte en la especificación funcional que se usa en el diseño físico. El diseño lógico es independiente de la tecnología. El modelo lógico refina, organiza y detalla la solución de negocios y define formalmente las reglas y políticas específicas de negocios.

El modelo lógico es el proceso de construir un esquema de la información que utiliza la empresa, basándose en un modelo de base de datos específico, independiente del SGBD concreto que se vaya a utilizar y de cualquier otra consideración física.

4.1.1. Obtención del modelo lógico-global de los datos a partir del DER

Para obtener el modelo lógico global de los datos, según el enfoque relacional a partir del **DER**, se sigue un procedimiento que iremos describiendo paso a paso y aplicándolo, así mismo, al ejemplo siguiente:

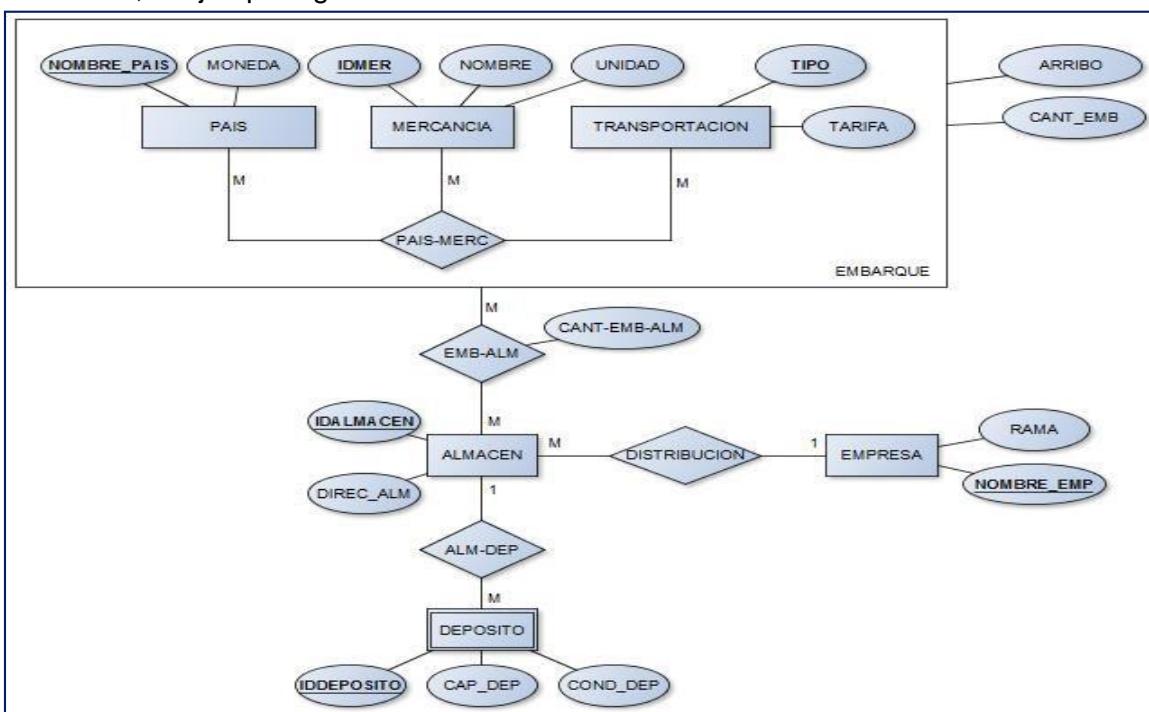


Figura 33: DER Movimiento Mercantil

Fuente. - Tomado desde la aplicación yEd

En el DER anterior se representa el movimiento mercantil de un organismo. En el organismo existen mercancías de las que se conoce su código, nombre y unidad de medida. Las mercancías proceden de diferentes países de los que se sabe nombre y tipo de moneda. Para la transportación de las mercancías existen diversas formas, cada una de las cuales se caracteriza por su tipo (barco, avión, tren, etc.) y tarifa.

Para cada mercancía de diferentes países existen diferentes formas de transportación; para cada país existen diferentes mercancías que son transportadas en diferentes formas de transportación; y una forma de transportación puede serlo de diferentes mercancías de diferentes países. Una mercancía procedente de un país y transportada de una forma dada constituye un embarque, y para éste se conoce su fecha de arribo y cantidad. Un embarque se distribuye entre diferentes almacenes y en un almacén se tienen diferentes embarques, cada uno en cierta cantidad. De cada almacén se tiene su código y dirección. Un almacén envía sus productos a una sola empresa y cada empresa

recibe productos de diferentes almacenes. Una empresa se caracteriza por su nombre y rama económica.

Cada almacén tiene distintos depósitos subordinados. De cada depósito se conoce su número (que se puede repetir en diferentes almacenes), capacidad y condiciones técnicas.

Pasos:

1. Representar las entidades regulares

PAÍS	(<u>NOMBRE_PAÍS</u> , MONEDA)
MERCANCÍA	(<u>IDMER</u> , NOMBRE, UNIDAD)
TRANSPORTACIÓN	(<u>TIPO</u> , TARIFA)
ALMACÉN	(<u>IDALMACEN</u> , DIREC_ALM)
EMPRESA	(<u>NOMBRE_EMP</u> , RAMA)

2. Representar las entidades agregadas

Representar, en una tabla relacional, cada entidad agregada con sus correspondientes atributos (entre ellos un identificador si fue definido) y las llaves de las entidades que forman la agregación. La llave primaria de esta tabla es la llave de la relación que da origen a la agregación si no hay un identificador definido; pero si lo hay, entonces la llave será dicho identificador

EMBARQUE (NOMBRE_PAÍS, IDMER, TIPO, ARRIBO, CANT_EMB)

Nótese que la llave estaría formada por las llaves de las 3 entidades regulares que intervienen en la agregación.

Pero, podía haberse definido un identificador para la entidad embarque. (Supóngase añadido en el DER un atributo de la entidad embarque, que sería su llave: IDEMBARQUE). Entonces se añadiría como atributo llave en la agregación y los 3 atributos NOMBRE_PAÍS, IDMER y TIPO permanecerían en la relación, pero no como llave así, por ejemplo:

EMBARQUE (IDEMBARQUE, NOMBRE_PAÍS, IDMER, TIPO,
ARRIBO, CANT_EMB)

3. Representar las entidades generalizadas y especializadas

Representar cada entidad generalizada en una tabla que contendrá sus atributos (sólo los de la generalizada) y, entre ellos, la llave.

Representar cada entidad especializada en una tabla que contendrá, como llave primaria, la llave de la generalización y los atributos propios sólo de la especialización.

Como este ejemplo no incluye ninguna generalización /especialización, analicemos cómo se llevaría a tablas el siguiente DER, en que se tiene la entidad generalizada estudiante y los casos especiales, becario y practicante.

4. Finalmente, las entidades resultantes son:

PAÍS	(<u>NOMBRE_PAÍS</u> , MONEDA)
MERCANCÍA	(<u>IDMER</u> , NOMBRE, UNIDAD)
TRANSPORTACIÓN	(<u>TIPO</u> , TARIFA)
EMPRESA	(<u>NOMBRE_EMP</u> , RAMA)
ALMACÉN	(<u>IDALMACEN</u> , DIREC_ALM, NOMBRE_EMP)
DEPOSITO	(<u>IDALMACEN</u> , <u>IDDEPOSITO</u> , CAP_DEP, COND_DEP)
EMBARQUE	(<u>NOMBRE_PAÍS</u> , <u>IDMER</u> , <u>TIPO</u> , ARRIBO, CANT_EMB)
EMBARQUE-ALMACEN	(<u>NOMBRE_PAÍS</u> , <u>IDMER</u> , <u>TIPO</u> , <u>IDALMACEN</u> , CANT_EMB_ALM)

5. El diagrama de base de datos en SQL es el siguiente:

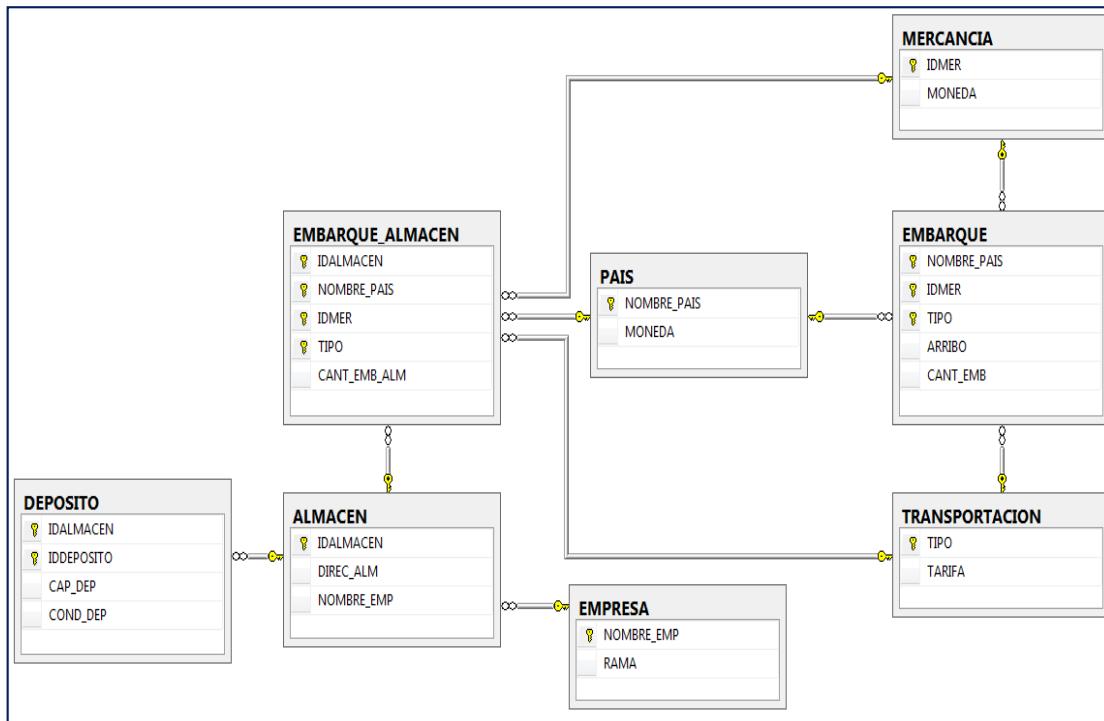


Figura 34: Diagrama de base de datos Movimiento Mercantil implementado en SQL

Fuente. - Tomado desde SQL Server 2014

Script en SQL Server 2014

```

USE MASTER
GO

IF DB_ID('MERCANTIL') IS NOT NULL
    DROP DATABASE MERCANTIL
GO

CREATE DATABASE MERCANTIL
GO

USE MERCANTIL
GO

CREATE TABLE PAIS (
    NOMBRE_PAIS      VARCHAR (30) NOT NULL PRIMARY KEY,
    MONEDA          VARCHAR (30) NOT NULL
)
GO

CREATE TABLE MERCANCIA (
    IDMER           CHAR (5) NOT NULL PRIMARY KEY,
    MONEDA          VARCHAR (30) NOT NULL
)
GO

CREATE TABLE TRANSPORTACION (
    TIPO            VARCHAR (30) NOT NULL PRIMARY KEY,
    TARIFA          MONEY      NOT NULL
)
    
```

```

)
GO
CREATE TABLE EMPRESA (
    NOMBRE_EMP      VARCHAR (30) NOT NULL PRIMARY KEY,
    RAMA            VARCHAR (30) NOT NULL
)
GO

CREATE TABLE ALMACEN (
    IDALMACEN       INT          NOT NULL PRIMARY KEY,
    DIREC_ALM      VARCHAR (50) NOT NULL,
    NOMBRE_EMP      VARCHAR (30) NOT NULL REFERENCES EMPRESA
)
GO

CREATE TABLE DEPOSITO (
    IDALMACEN       INT          NOT NULL REFERENCES ALMACEN,
    IDDEPOSITO     VARCHAR (50) NOT NULL,
    CAP_DEP         DECIMAL (8,2) NOT NULL,
    COND_DEP        VARCHAR (50) NOT NULL,
    PRIMARY KEY (IDALMACEN, IDDEPOSITO)
)
GO

CREATE TABLE EMBARQUE (
    NOMBRE_PAIS     VARCHAR (30) NOT NULL REFERENCES PAIS,
    IDMER           CHAR (5) NOT NULL REFERENCES MERCANCIA,
    TIPO            VARCHAR (30) NOT NULL REFERENCES TRANSPORTACION,
    ARRIBO          VARCHAR (50) NOT NULL,
    CANT_EMB        INT          NOT NULL,
    PRIMARY KEY (NOMBRE_PAIS, IDMER, TIPO)
)
GO

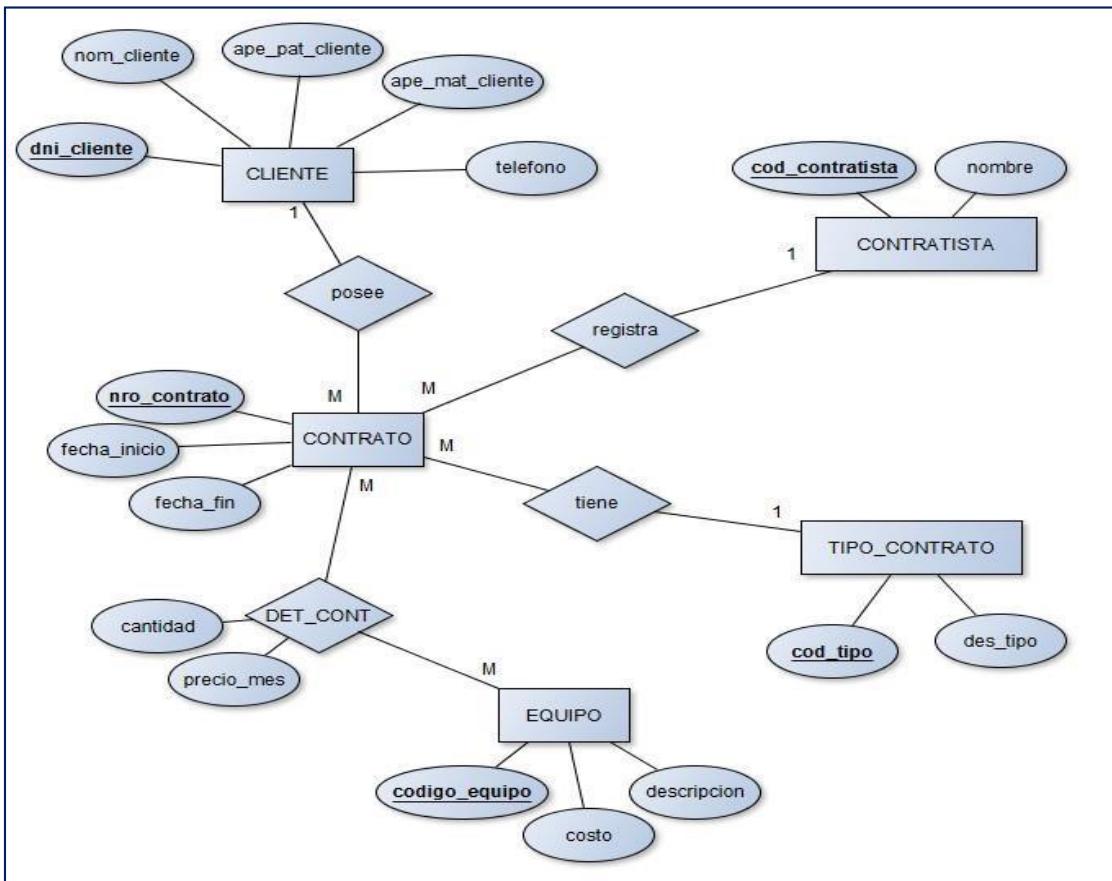
CREATE TABLE EMBARQUE_ALMACEN (
    IDALMACEN       INT          NOT NULL REFERENCES ALMACEN,
    NOMBRE_PAIS     VARCHAR (30) NOT NULL REFERENCES PAIS,
    IDMER           CHAR (5) NOT NULL REFERENCES MERCANCIA,
    TIPO            VARCHAR (30) NOT NULL REFERENCES TRANSPORTACION,
    CANT_EMB_ALM   INT          NOT NULL,
    PRIMARY KEY (IDALMACEN, NOMBRE_PAIS, IDMER, TIPO)
)
GO

```

4.1.2. Modelo lógico. Ejercicios

Ejercicio Propuesto 01: CONTRATO DE EQUIPOS

Implemente en SQL Server 2014 el siguiente modelo DER



Nota: Obtenga el modelo lógico-global de los datos a partir del DER que representa cada uno de los siguientes ejercicios:

Ejercicio propuesto 02: INDUSTRÍA MECÁNICA

En una empresa de la industria mecánica se fabrican distintos tipos de piezas. Para cada una se conoce el código que la identifica, descripción y peso unitario. Sobre cada pieza se realizan distintas operaciones (corte, fresado, etc.) acerca de las cuales se conoce su código y nombre. En su proceso de fabricación, una pieza pasa por diferentes equipos que pueden realizar diferentes operaciones, una operación puede ser realizada en diferentes equipos a diferentes piezas y en un equipo se realizan diferentes operaciones sobre diferentes piezas. De cada equipo se conoce su nombre, modelo y país de procedencia. Una operación realizada sobre una pieza en un equipo dado constituye una norma que se identifica por un código y tiene asociados una descripción y un tiempo de ejecución.

En la empresa trabajan obreros de dos tipos distintos: directos y auxiliares. Un obrero directo está vinculado con el cumplimiento de diferentes normas y una norma puede ser aplicada a distintos obreros directos. Un obrero directo cumple una norma dada en un determinado por ciento. Un obrero auxiliar realiza el mantenimiento de diferentes equipos y un equipo dado siempre es atendido por un mismo obrero auxiliar. Para cada equipo atendido por un obrero auxiliar dado se conoce el tipo de mantenimiento efectuado y la última fecha en que se realizó. Todos los obreros de la fábrica están caracterizados por un número, su nombre, calificación laboral y categoría ocupacional (directo o auxiliar).

Ejercicio propuesto 03: REPARACIÓN DE RADIOS

Se desea controlar la actividad de reparación de radios en los talleres dedicados a esta tarea. Cada usuario puede llevar diferentes radios para ser reparados, aunque un radio sólo pertenece a un usuario. Para cada radio se conoce el número de la solicitud de reparación que lo identifica, tipo de radio, descripción y tipo de rotura (**sencillas** o **complejas**). De cada usuario se conoce su DNI, nombre y dirección. Cada técnico de reparaciones pertenece a un taller y tiene un código que lo identifica dentro de su taller, por lo que cada código puede repetirse para diferentes talleres, y tiene, además, un nombre, una categoría (**A**, **B**, **C**) y un salario. En un taller laboran muchos técnicos de reparaciones. De cada taller se conoce el código que lo identifica, nombre y dirección. Los radios con roturas **complejas** sólo podrán ser atendidos por técnicos de categoría **A**, mientras que los radios con roturas **sencillas** pueden ser atendidos por cualquier técnico. En cualquier caso, un técnico puede reparar muchos radios, pero un radio es reparado por un solo técnico.

Ejercicio propuesto 04: EMPRESA DE “TAXIS”

En una empresa de taxis de Lima existe un conjunto de taxis que brindan servicio a los hospitales de dicha ciudad. En la empresa trabajan choferes, cada uno de los cuales se caracteriza por su DNI, nombre y años de servicio. En la empresa hay numerosos taxis, de los que se conoce la placa, el número del motor, la marca y el modelo. Un taxi puede ser conducido por diferentes choferes (en distintos momentos), pero un chofer siempre conduce el mismo taxi. Se conoce la cantidad de kilómetros totales recorridos por un chofer en su correspondiente taxi. La empresa brinda servicios a varios hospitales. De cada hospital se conoce su nombre, tipo y dirección. Un chofer le brinda servicios (realiza viajes) a distintos hospitales y a un hospital le brindan servicios distintos choferes. Se sabe la cantidad de viajes realizados por cada chofer a cada hospital.

Ejercicio propuesto 05: INSTITUTO AERONÁUTICA DEL PERÚ

El Instituto de Aeronáutica del Perú cuenta con información sobre las diferentes compañías de aviación que existen en el mundo. De cada compañía se conoce su nombre (Cubana, Copa, AeroContinente, etc.), su volumen anual de ventas y un código que la identifica. Cada compañía puede estar representada en diferentes países y en un país pueden estar representadas diversas compañías. De cada país se conoce su código, nombre, idioma y área de moneda. También se sabe que por vía aérea se realizan diferentes viajes. Cada viaje posee un código, un lugar de origen, un destino y una cantidad de kilómetros a recorrer. Además, se conoce que existen distintos tipos de aviones. Cada tipo de avión se identifica por un nombre (IL-62, DC-10, etc.) y posee un consumo de gasolina por kilómetro y una cierta cantidad de asientos. Un mismo viaje puede ser realizado por distintos tipos de aviones y un tipo de avión puede ser utilizado en diversos viajes. Un viaje realizado por un tipo de avión constituye un vuelo y, para cada vuelo, se conoce su tiempo de duración.

Una compañía de aviación realiza muchos vuelos y un mismo vuelo puede ser cubierto por diversas compañías. Para cada vuelo que ofrece una compañía se conoce el precio del pasaje.

Ejercicio propuesto 06: FARMACIA

Se desea mantener una base de datos para una cadena de farmacias distribuida en diferentes ciudades. Cada farmacia tiene sus empleados propios y un farmacéutico. Por

cada ciudad existe un único farmacéutico; esto es, si en una ciudad hubiera más de una farmacia, el mismo farmacéutico estaría afectado a todas las farmacias de esa ciudad. Cada farmacia tiene a su vez su stock de medicamentos. El mismo se mantiene por medicamento y presentación. Los medicamentos se organizan según la o las monodrogas que lo componen, su presentación (por ejemplo, ampollas de 5 unidades, jarabe de 100ml, inyecciones por 10 unidades, pomada 60gr, etc.), el laboratorio que lo comercializa, y su acción terapéutica (analgésico, antibiótico, etc.). Por cada medicamento se mantiene su precio y la cantidad en existencia del mismo. El sistema deberá permitir consultar la base de datos de diferentes alternativas para medicamentos compuestos por una monodroga, medicamentos de un laboratorio, presentaciones de un medicamento, entre otras. Se pide:

- a) Construir el esquema entidad/relación añadiendo el mínimo número de atributos, imponiendo y explicando las restricciones de cardinalidad mínimo-máximo, sin considerar los costes de las operaciones, pero asegurando que se puedan realizar.

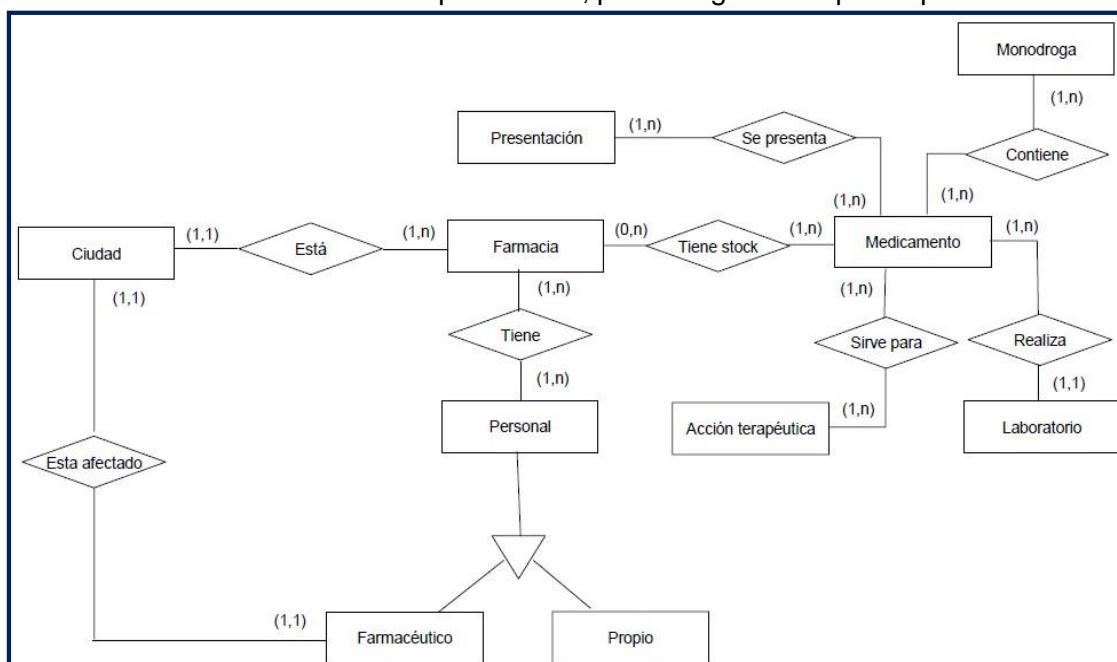


Figura 35: Diagrama Entidad Relación Farmacia

Fuente. - Tomado desde <https://www.fdi.ucm.es/profesor/fernand/DBD/examenjunio04.pdf>

Ciudad (#nombre, dni_farmacéutico)
 Farmacia (#número, nombre, dirección, nombre_ciudad)
 Personal (#DNI, nombre)
 Farmacéutico (#DNI, fechatítulo, nombre_ciudad)
 Propio (#DNI, salario)
 Medicamento (#código, nombre, precio, cantidad, código_laboratorio)
 Acción_terapéutica (#tipo_acción, descripción)
 Laboratorio (#código, nombre)
 Presentación (#tipo_presentación)
 Monodroga (#código, descripción)
 Tiene (#número_farmacia, #dni_personal)
 Tiene_stock (#número_farmacia, #código_medicamento)
 Se presenta (#código_medicamento, #tipo_presentación)

Contiene (#código_medicamento, #código_monodroga)

Sirve para (#código_medicamento, #tipo_acción)

Ejercicio propuesto 06: PRODUCCIÓN DE MEDICAMENTOS

Diseñe el modelo de datos relacional a partir del siguiente Diagrama Entidad-Relación que representa el de la producción de medicamentos en distintas formas de presentación (cápsulas, ampollas, etc.) y su distribución a los distintos destinos que tienen estos medicamentos. El atributo tipodest indica si el destino es una farmacia o un hospital.

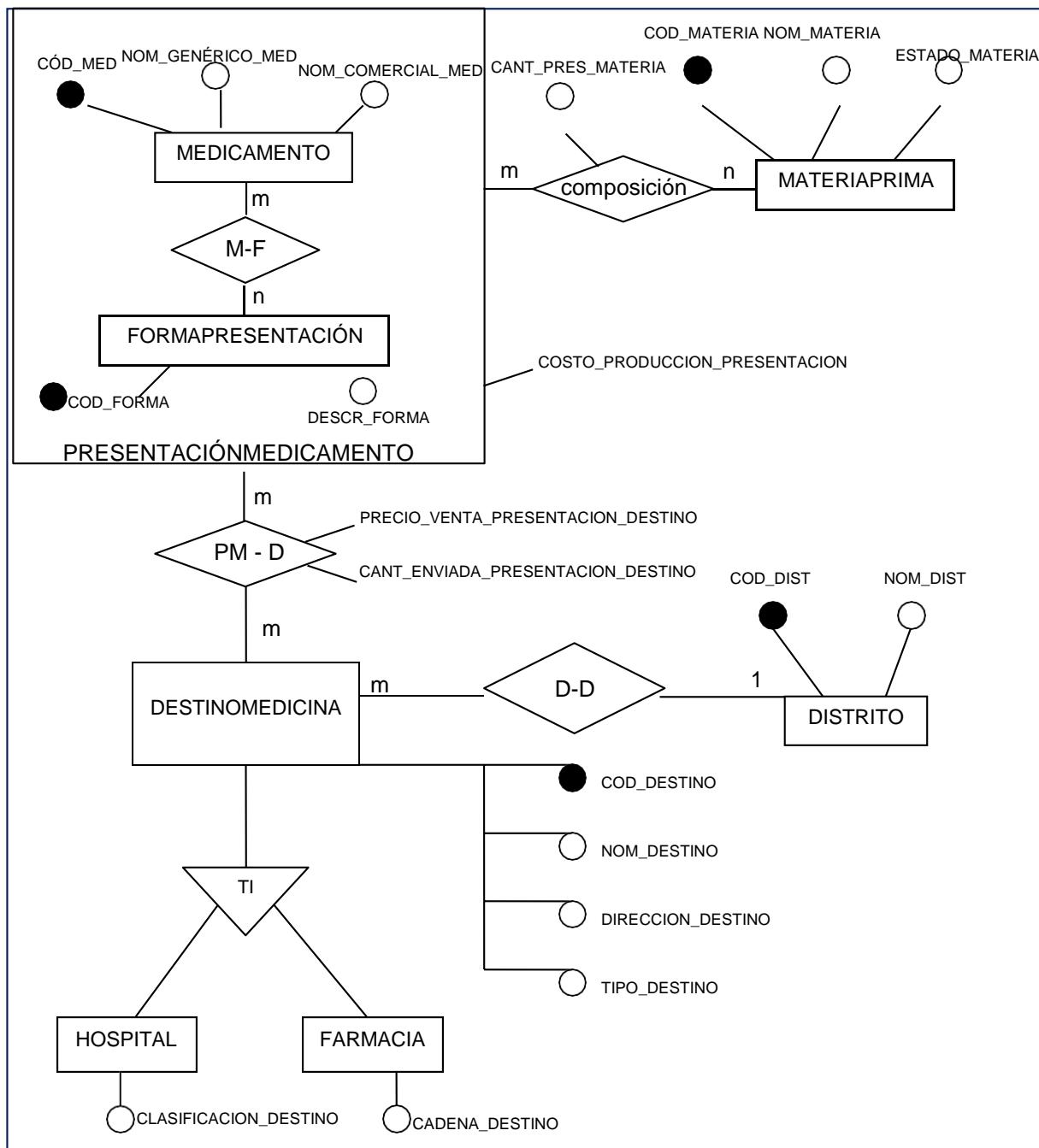


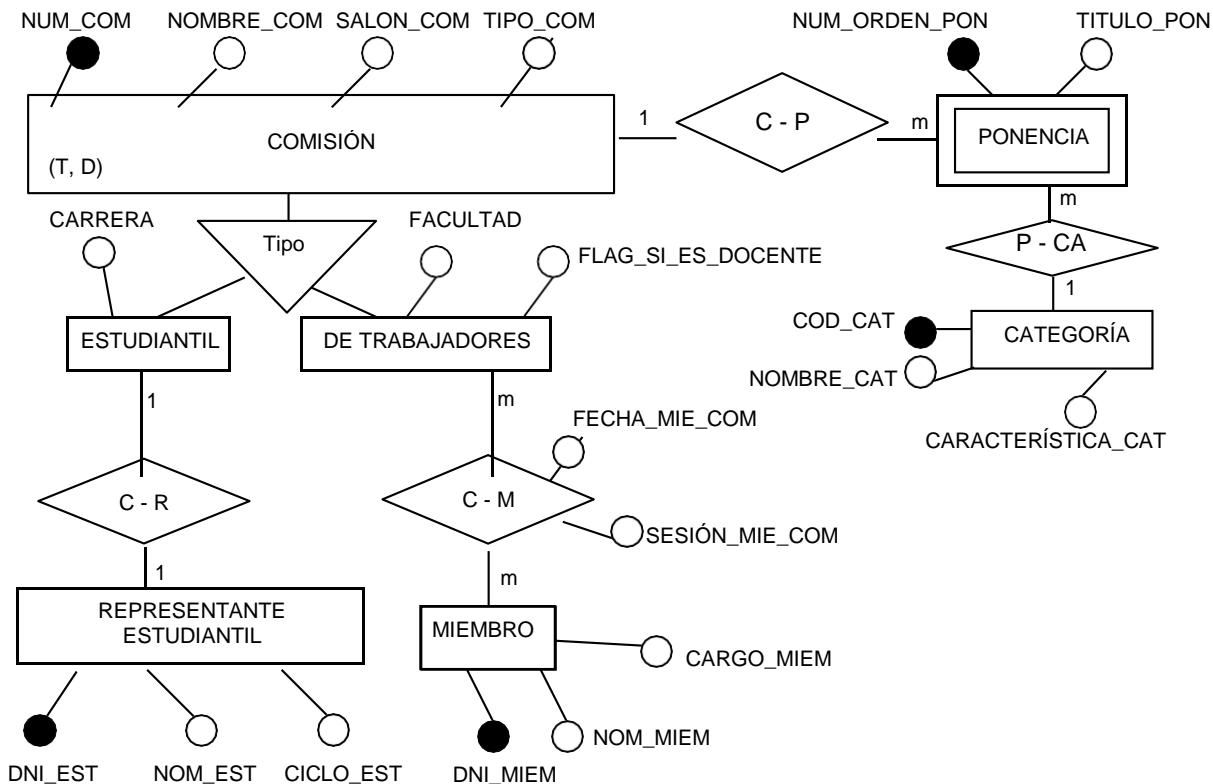
Figura 36: Diagrama Entidad Relación Producción de medicamentos

Fuente. - Tomado desde

http://informatica.cubaeduca.cu/index.php?option=com_content&view=category&id=432&Itemid=101&limitstart=48

Ejercicio propuesto 07: EVENTO CIENTÍFICO

Diseñe el modelo de datos relacional a partir del siguiente Diagrama Entidad-Relación que representa la organización de un evento científico en una universidad, en el que participan estudiantes y trabajadores (que pueden ser docentes y no docentes).



Ejercicio propuesto 08: MANTENIMIENTO DE EQUIPOS

En un taller de mantenimiento de equipos de computación se desea controlar los servicios brindados. En el taller laboran varios trabajadores. De cada uno se conoce su DNI, su nombre, categoría laboral y tipo de mantenimiento (**Tipo1**, **Tipo2**) que se encarga de realizar a los equipos. El mantenimiento de los equipos de computación se ejecuta por órdenes de servicio que son presentadas por las empresas usuarias de este taller. Una empresa usuaria puede presentar muchas órdenes de servicio y cada orden de servicio es presentada por una empresa usuaria. De cada orden de servicio se conoce el número que la identifica dentro de la empresa que la presenta, por lo que puede repetirse ese número para diferentes empresas y la fecha en que aparece. De cada empresa usuaria se sabe el código que la identifica, su nombre y la rama económica a la que pertenece. Una orden de servicio puede contener la solicitud de mantenimiento de varios equipos y puede solicitarse el mantenimiento de un equipo en diferentes órdenes de servicio. De cada equipo se sabe el código que lo identifica, su descripción, su marca y su tipo (sólo se toman en cuenta dos tipos: **Computadoras** y **Otros**). El mantenimiento de las **Computadoras** sólo puede ser realizado por los trabajadores que realizan mantenimientos **Tipo1** y el mantenimiento de los **Otros** equipos sólo puede realizarse por trabajadores encargados de hacer los mantenimientos **Tipo2**. En ambos casos, el mantenimiento de un equipo es realizado

por un solo trabajador, aunque un trabajador puede realizar el mantenimiento de varios equipos.

Ejercicio propuesto 09: LOCOMOTORAS

Se desea controlar la programación de los itinerarios a recorrer por las locomotoras en una estación de ferrocarriles. En la estación existen varias locomotoras, de las que se conoce el código que la identifica y años que lleva prestando servicios. Existen diversas marcas de locomotora; una locomotora es de una marca y una marca puede serlo de varias locomotoras. De cada marca se tiene el código que la identifica, el consumo de combustible promedio por hora y la velocidad máxima que puede alcanzar. Una locomotora realiza diversos itinerarios, pero un itinerario lo realiza una sola locomotora. De cada itinerario se conoce el número que lo identifica para una locomotora dada, pero que puede repetirse para distintas locomotoras (es decir, la locomotora A puede tener un itinerario 2 y la locomotora B puede tener también un itinerario 2), los kilómetros a recorrer, el destino final y el tipo de itinerario (**Diurno o Nocturno**; un itinerario es diurno o es nocturno y todos los itinerarios son de uno de estos dos tipos). Las locomotoras son conducidas por maquinistas. Un maquinista (cualquiera) puede conducir varias locomotoras y una locomotora puede ser conducida por muchos maquinistas y se conoce la cantidad de kilómetros recorrido por un maquinista en una locomotora. De cada maquinista se conoce su DNI, su nombre y si tiene más de 10 años de antigüedad como maquinista o no. Los itinerarios nocturnos sólo los pueden realizar los maquinistas con más de 10 años de antigüedad y los maquinistas con más de 10 años de antigüedad sólo realizan itinerarios nocturnos. Los itinerarios diurnos sólo los realizan los maquinistas con 10 o menos años de antigüedad y los maquinistas con 10 o menos años de antigüedad sólo realizan itinerarios diurnos. En ambos casos ocurre que un maquinista realiza muchos itinerarios y que un itinerario es realizado por muchos maquinistas.

4.2. Tema 11: Modelo Físico (MER a DER)

Recibe como entrada el esquema lógico y da como resultado un esquema físico, que es una descripción de la implementación de una base de datos en la memoria secundaria, describe las estructuras de almacenamiento y los métodos usados para tener un acceso efectivo a los datos.

Modelos Físicos: Modelo Unificador, Memoria de Elementos.

Estructura de diseño de base de datos

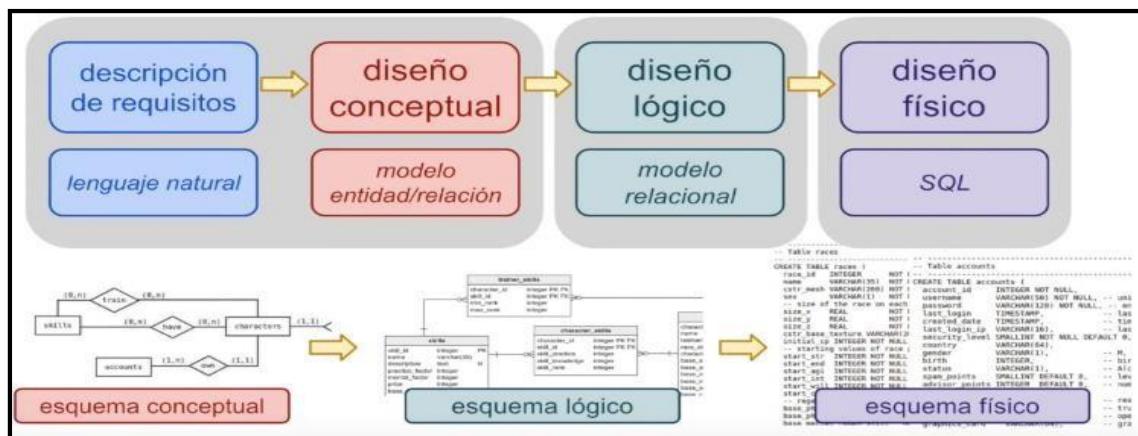


Figura 37: Estructura de diseño de base de datos

Fuente. - Tomado desde <https://www.youtube.com/watch?v=21q2XhbHmNc>

4.2.1. Metodología de sistema de base de datos

La metodología para el diseño de la base de datos, consta de los siguientes pasos:

- Determinación de entidades y atributos
- Normalización de entidades
- Determinación de relaciones (DER)
- Obtención del modelo lógico global de los datos
- Diseño físico de la base de datos

Cuando se va a realizar el diseño de la base de datos para un sistema dado, es necesario determinar los datos que se deben tomar en cuenta y las dependencias funcionales existentes entre ellos.

Rigurosamente, esto se obtiene luego de realizada la etapa de análisis del sistema y partiendo de lo obtenido en ésta, pero trataremos de dar indicaciones que permitan lograr un buen diseño de la base de datos a partir de lo que conocemos.

La situación a partir de la cual se discutirá la metodología para diseñar la base de datos es la siguiente:

En un hospital Clínico Quirúrgico se desea automatizar el control de la actividad quirúrgica. En el hospital se tienen varios quirófanos y se realizan intervenciones quirúrgicas en diferentes especialidades. Las intervenciones quirúrgicas pueden ser electivas (planificadas) o urgentes (no planificadas). Se tiene la planificación de las operaciones quirúrgicas a realizar cada día. Las operaciones planificadas pueden ser suspendidas por distintas causas. Para cada operación realizada (haya sido planificada o no), se confecciona un informe operatorio. En cada operación realizada se emplean materiales cuyo gasto se controla.

Las salidas o reportes que se quieren obtener con el sistema automatizado son las siguientes:

PROGRAMACIÓN QUIRÚRGICA					Para el día: ____ de ____ del 20 ____			
Paciente	Hist Clín.	Edad	Sexo	Diagnóstico Pre-Oper.	Intervenc. Indicada	Hora	Salón	Cirujano

OPERACIONES SUSPENDIDAS			En el día: ____ de ____ del 20 ____	
Paciente	Historia Clínica	Edad	Causa	

INFORME OPERATORIO		Fecha de Operación:	
		Hora Comienzo	Hora Término
Paciente		Historia Clínica	Edad
Diagnóstico Clínico: _____ Operación Realizada: _____ Diagnóstico Operatorio: _____			
Situación Final del Paciente: Satisfactoria <input type="checkbox"/> No Satisfactoria <input type="checkbox"/> Fallecido <input type="checkbox"/>			
Especialidad		Tipo de Operación Realizada Electiva <input type="checkbox"/> Urgente <input type="checkbox"/>	
Cirujano: _____			

RESUMEN DE ACTIVIDADES QUIRÚRGICAS		Mes: _____
Cantidad de Operaciones Electivas: _____		
Cantidad de Operaciones Urgentes: _____		
Especialidad	Cantidad de Operaciones	

GASTO DE MATERIALES DE OPERACIÓN						En el día: ____de ____ del 20____		
Paciente	Hist. Clín.	Edad	Hora Com.	Patología	Cód. Mat.	Descr. Mater.	Cantidad	Costo

Para comenzar a estudiar la metodología de diseño de la base de datos, partiremos de una premisa: lo que se obtiene con un sistema automatizado son sus salidas, es decir, el resultado que se desea obtener. Lo que llega al usuario son, justamente, las salidas o reportes del sistema y para ello se desarrolla. Por lo tanto, lo que no sea necesario para obtener las salidas del sistema, no tiene por qué estar almacenado en la base de datos. En fin, lo que debe almacenarse en la base de datos es lo que es imprescindible para obtener las salidas.

Es por esta razón que, para el diseño de la base de datos, se parte del análisis de las salidas que se desea obtener con el sistema.

4.2.1.1. Determinación de entidades y atributos

Para cada salida:

- Consultar su formato
- Determinar datos que se calculen u obtengan a partir de otros e ir sustituyéndolos hasta llegar a los primarios
- Determinar la existencia de archivos con información normativa y/o de consulta
- Cada salida y cada archivo, tomarlos como entidades y relacionar (listar) sus atributos

En el ejemplo, representando cada una de las salidas por una relación, se tienen las siguientes relaciones o entidades iniciales:

PQ (Fecha, NomPac, HC, Ed, Sexo, DiagPre, Interv, Hora, Salón, Cirujano)

OS (Fecha, NomPac, HC, Ed, Causa)

IO (Fecha, HoraCom, HoraTerm, NomPac, HC, Ed, DiagClín, Operac, DiagOper, SitPac, Espec, TipoOpReal, CirReal)

RAQ (Mes, Elect, Urg, Espec, Cant)

GMO (Fecha, NomPac, HC, Ed, HoraCom, Patología, CódMat, DescMat, Cant, Costo)

Analizando cada dato se ve que:

En **PQ**:

Hora se calcula según el tiempo promedio de demora para cada operación, por lo que es necesario un archivo normativo:

⇒ **TPO** (Operación, Tiempo Prom)

Salón se decide teniendo en cuenta la distribución de los salones, que, en este caso, es por día de la semana:

⇒ **DS** (Salón, DíaSem, Espec)

En **OS**:

Todos son datos primarios

En **IO**:

Todos son datos primarios

En **RAQ**:

Elect, *Urg* y *Cant* son datos que se calculan a partir de contar en **IO**, por lo que la relación **RAQ** se puede eliminar completamente para el análisis, pues al ser

calculables todos los datos contenidos en ella, no aporta información nueva a almacenar en la base de datos.

En **GMO**:

$\text{Costo} = \text{cant} \times \text{precio}$, por lo que se sustituye costo por precio (ya cant está en el modelo)

Esto implica la necesidad de que exista un archivo de consulta MATERIAL (puede salir en entrevistas a usuarios)

4.2.1.2. Normalización de entidades

Para cada relación obtenida en el paso 1:

- Determinar las DF
- Normalizar cada entidad
- Fusionar, de ser lógico, las entidades normalizadas que tengan la misma llave

Entonces, en el ejemplo, tendríamos:

PQ (Fecha, NomPac, HC, Ed, Sexo, DiagPre, Interv, Hora, Salón, Cirujano)

- **Normalizando:**

Paciente (HC, NomPac, Ed, Sexo)

PQ (Fecha, HC, DiagPre, Interv, Hora, Salón, Cirujano)

OS (Fecha, NomPac, HC, Ed, Causa)

- **Normalizando:**

Paciente1 (HC, NomPac, Ed)

OS (Fecha, HC, Causa)

IO (Fecha, HoraCom, HoraTerm, NomPac, HC, Ed, DiagClín, Operac, DiagOper, SitPac, Espec, TipoOpReal, CirReal)

- **Normalizando:**

2FN:

Paciente2 (HC, NomPac, Ed)

PlanOperac (Fecha, HC, DiagClín, Operac, Espec)

IO (Fecha, HoraCom, HC, HoraTerm, DiagOper, SitPac, TipoOpReal, CirReal, OperReal)

3FN:

Paciente2 (HC, NomPac, Ed)

PlanOperac (Fecha, HC, DiagClín, Operac)

ClasifOperac (Operac, Espec)

IO (Fecha, HoraCom, HC, HoraTerm, DiagOper, SitPac, TipoOpReal, CirReal, OperReal)

RAQ (ya está eliminada, como se vio antes, para este análisis)

GMO (Fecha, NomPac, HC, Ed, HoraCom, Patología, CódMat, DescMat, Cant, Precio)
 (Precio en lugar de Costo)

- **Normalizando:**

Paciente3 (HC, NomPac, Ed)

Diagnóstico (Fecha, HC, Patología)

Material (CódMat, DescMat, Precio)

GMO (Fecha, HC, HoraCom, CódMat, Cant)

Además, se tienen los dos archivos normativos o de consulta:

TPO (Operación, TiempoProm)

DS (Salón, DíaSem, Espec)

Fusionando las entidades normalizadas con igual llave, siempre que sea lógico, se tiene:

De **Paciente**, **Paciente1**, **Paciente2** y **Paciente3**:

Paciente (HC, NomPac, Ed, Sexo)

De **PQ**, **OS**, **PlanOperac** y **Diagnóstico**:

PlanOperac está contenida en **PQ** y **Diagnóstico** está contenida en **PQ** también. Tener en cuenta que DiagPre = DiagClín = Patología son alias diferentes para indicar un mismo atributo, que tiene el mismo dominio.

Pero **OS**, a pesar de tener igual llave, está representando otra entidad, la **Operación Suspendida**; semánticamente, no representa lo mismo que una **Operación Planificada** y, por lo tanto, no tiene sentido unirlas. Es más, si se hiciera, todas las operaciones tendrían el atributo Causa, y en la mayoría de las ocurrencias, este atributo estaría vacío, pues sólo son algunas operaciones las que se suspenden. Por lo tanto, de **PQ**, **OS**, **PlanOperac** y **Diagnóstico** quedarían:

PQ (Fecha, HC, DiagPre, Interv, Hora, Salón, Cirujano)

OS (Fecha, HC, Causa)

De **ClasifOperac** y **TPO**:

Intervención (Operación, TiempoProm, Espec)

De **DS**:

DS (Salón, DíaSem, Espec)

De **IO**:

IO (Fecha, HoraCom, HC, HoraTerm, DiagOper, SitPac, TipoOpReal, CirReal, OperReal)

De **Material**:

Material (CódMat, DescMat, Precio)

De **GMO**:

GMO (Fecha, HC, HoraCom, CódMat, Cant)

Por lo que, en este paso, finalmente, quedarían las siguientes relaciones:

Paciente (HC, NomPac, Ed, Sexo)

PQ (Fecha, HC, DiagPre, Interv, Hora, Salón, Cirujano)

OS (Fecha, HC, Causa)

Intervención (Operación, TiempoProm, Espec)

DS (Salón, DíaSem, Espec)

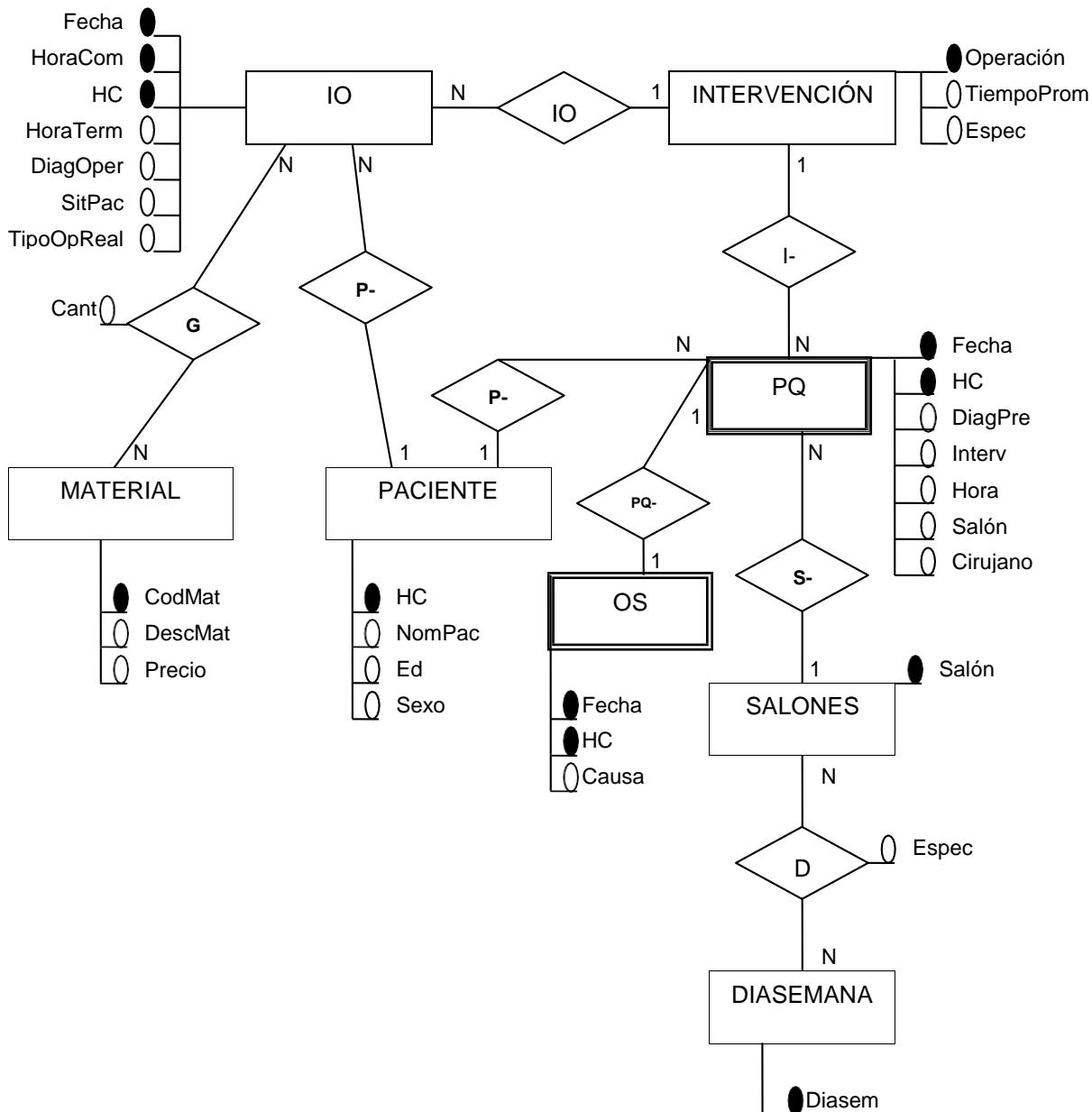
IO (Fecha, HoraCom, HC, HoraTerm, DiagOper, SitPac, TipoOpReal, CirReal, OperReal)

Material (CódMat, DescMat, Precio)

GMO (Fecha, HC, HoraCom, CódMat, Cant)

4.2.1.3. Determinación de relaciones (DER)

- ✓ Analizar cada entidad con las restantes para determinar si existe relación, determinando tipo de proyección
- ✓ Si hay alguna entidad cuya llave es compuesta, de modo que resulta que esa llave es la combinación de las llaves de otras entidades, es porque representa una relación entre ellas de m: n
- ✓ Si hay alguna entidad cuya llave es compuesta y un subconjunto de los atributos que la forma no es llave de otra entidad, es porque es una entidad débil
- ✓ Si en una entidad aparece, como atributo no llave, la llave de otra entidad es porque existe entre ellas una relación de m:1
- ✓ Analizar entidades agregadas, entidades generalizadas/especializadas y sus relaciones con otras
- ✓ Ir confeccionando el DER
- ✓ Fusionar, de ser lógico, las entidades normalizadas que tengan la misma llave



En este paso se puede comprobar si lo que se ha realizado es correcto o, si surge alguna contradicción, se puede completar el modelo con nuevos elementos que se hayan podido obtener en intercambios con los usuarios pues con el DER se tiene un modelo menos abstracto, más cercano. El analista puede proponer nuevas posibilidades a los usuarios y agregarlas en caso de acordarse esto, etc.

4.2.1.4. Obtención del modelo lógico global de los datos

Aquí se deben seguir los pasos para llevar del DER al modelo lógico global de los datos. Si en el paso anterior no se modifica el DER obtenido, entonces las tablas que se obtienen en este paso, en general, son las mismas que las obtenidas en el segundo paso (Normalizar las entidades). Supongamos que, en nuestro ejemplo, éste es el caso. Entonces:

Diseño lógico:

Paciente (HC, NomPac, Ed, Sexo)

Intervención (operación, TiempoProm, Espec)

Material (CódMat, DescMat, Precio)

PQ (Fecha, HC, DiagPre, Interv, Hora, Salón, Cirujano)

IO (Fecha, HoraCom, HC, HoraTerm, DiagOper, SitPac, TipoOpReal, CirReal, OperReal)

OS (Fecha, HC, Causa)

DS (Salón, DíaSem, Espec)

GMO (Fecha, HC, HoraCom, CódMat, Cant)

4.1.2.5. Diseño físico de la base de datos

En este paso hay que tener en cuenta:

- Aplicaciones a realizar sobre los datos
- Características particulares del SGBD

Ejemplos de estos dos aspectos a tener en cuenta:

- Considerar la inclusión de campos que sean secundarios debido a la cantidad de veces que habría que calcularlos en una determinada aplicación
- Separar un archivo en varios debido a la cantidad de campos y lo que se accede cada vez.
- archivos índices
- archivos para reportes

Desarrolle los siguientes ejercicios:

Suponiendo que, en cada caso, las tablas que se presentan han sido las únicas obtenidas después de concluirse el segundo paso de la metodología para el diseño de bases de datos (normalización), obtenga el diagrama entidad-relación correspondiente:

Caso 01

niño (idniño, nomniño, fechnac, sexo)

vacuna (códvac, nombrevac, paísproced)

vacunación (idniño, códvac, fechavacunación)

Caso 02

cigarro (nombrecigarro, calidad, preciounidad, códempresa)

empresaproductora (códempresa, nomempresa, añosdefundada)

Caso 03

trabajopresentadoaconcurso (códtrab, título trab, categoría, tipotrab)

trabajopremiado (códtrab, premioobtenido, fechapublicación)

trabajodestacado (códtrab, lugaralcanzado)

tipotrab, indica si el trabajo presentado a concurso fue premiado, destacado o no tuvo ninguna distinción

Caso 04

producto (codprod, nomprod, precio)

empresa (numemp, nmempresa, rama)

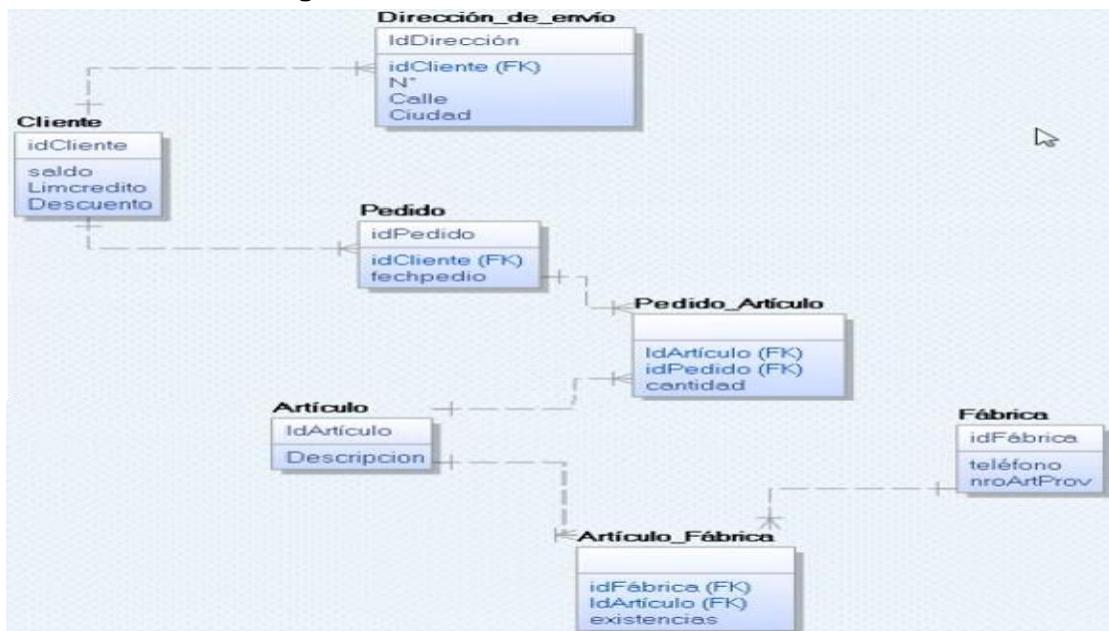
producción (numemp, codprod, cantidadproducida)

almacén (codalm, diralm, capacidad)

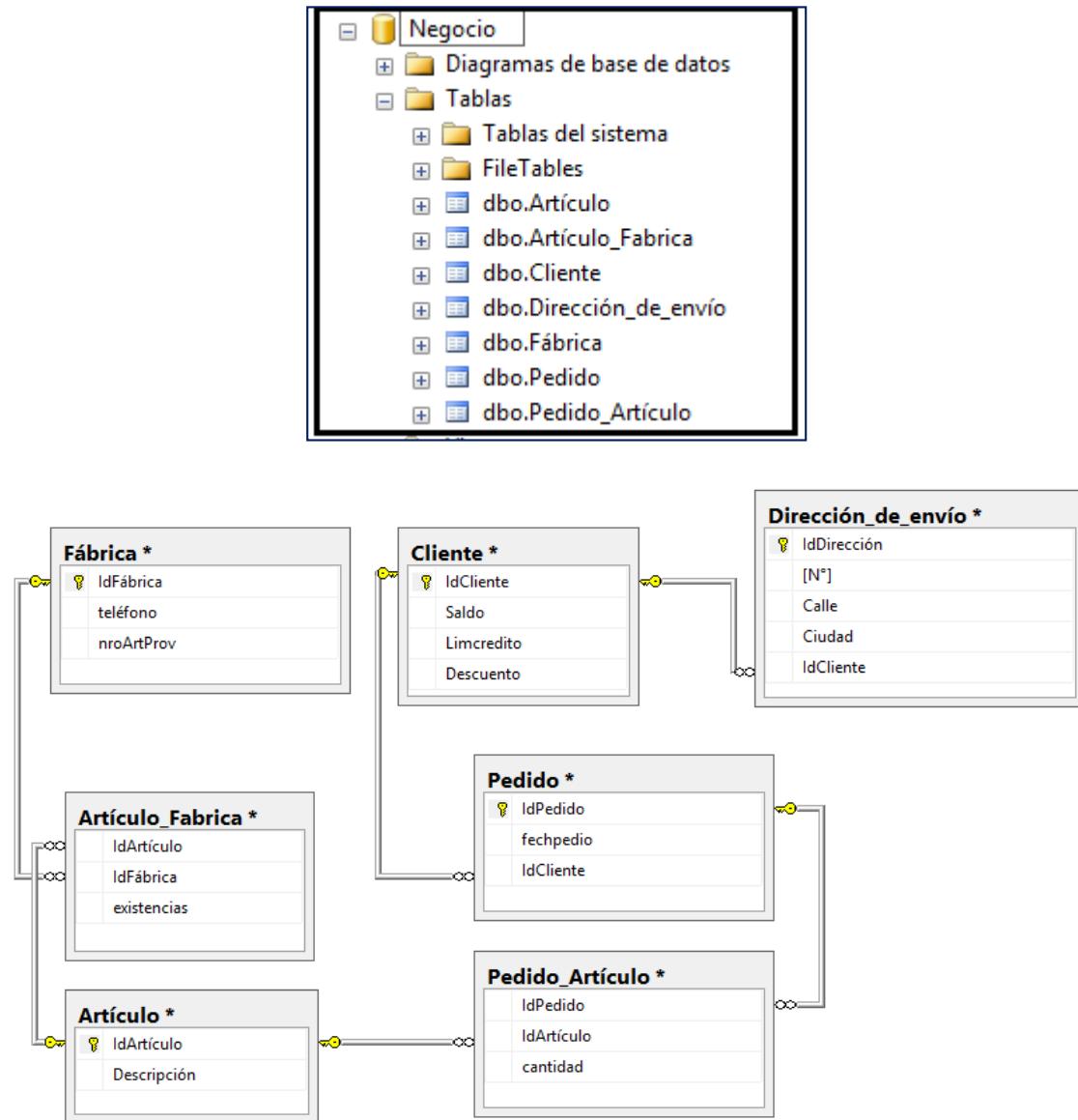
- Caso 05**
distribución (numemp, codprod, codalm, cantidadrecibida, fecharecepción)
- Caso 06**
grupo (numgrupo, ubicación)
alumno (numgrupo, numlista, nombrealumno, fechnac)
- Caso 07**
compañíaaviación (nomcomp, sedeprincipal, añosdefundada)
viaje (codviaje, origen, destino, cantidadkm)
vuelo (nomcia, codviaje, precio, frecuencia)
- Caso 08**
tipocent, indica si el centro cultural es un teatro, un museo o una sala de exposiciones
Caso 08
periódico (codperiod, nomperiod, país)
noticia (codperiod, fecha, numnoticia, título, textonoticia)
- Caso 09**
medicamento (codmed, nomgenérico, nomcomercial)
formadepresentación (codforma, descripforma)
medicamento_forma (codmed, codforma)
presentacióndelmedicamento (codmed, codforma, costoproducción)
materiaprima (codmat, nommat, estadofísico)
composición (codmed, codforma, codmat, cantidad)
- Caso 10**
temainvestigación (codtema, nomtema, ramacientífica, DNIprofesor)
profesor (DNIprofesor, nomprofesor, categoríaprofesor)

4.2.2. Obtención del modelo lógico-global de los datos a partir del MER

- **Modelo Lógico**



- Convertido en Modelo Físico



Resumen

||| La metodología para el diseño de la base de datos, consta de los siguientes pasos:

- Determinación de entidades y atributos
- Normalización de entidades
- Determinación de relaciones (DER)
- Obtención del modelo lógico global de los datos
- Diseño físico de la base de datos

||| Para la determinación de entidades y atributos, para cada salida, se hace lo siguiente:

- Consultar su formato
- Determinar datos que se calculen u obtengan a partir de otros e ir sustituyéndolos hasta llegar a los primarios
- Determinar la existencia de archivos con información normativa y/o de consulta
- Cada salida y cada archivo, tomarlos como entidades y relacionar (listar) sus atributos en una tabla

||| En la normalización de entidades, para cada tabla definida, hay que:

- Determinar las DF
- Normalizar cada entidad
- Fusionar, de ser lógico, las entidades normalizadas que tengan la misma llave

||| Determinación de relaciones (DER)

- Analizar cada entidad con las restantes para determinar si existe relación, determinando tipo de proyección.
- Si hay alguna entidad cuya llave es compuesta, de modo que resulta que esa llave es la combinación de las llaves de otras entidades, es porque representa una relación entre ellas de m: n
- Si hay alguna entidad cuya llave es compuesta y un subconjunto de los atributos que la forma no es llave de otra entidad, es porque es una entidad débil
- Si en una entidad aparece, como atributo no llave, la llave de otra entidad es porque existe entre ellas una relación de m:1
- Analizar entidades agregadas, entidades generalizadas/especializadas y sus relaciones con otras
- Ir confeccionando el DER
- Fusionar, de ser lógico, las entidades normalizadas que tengan la misma llave

||| Obtención del modelo lógico global de los datos se realiza de acuerdo a lo estudiado en las tres primeras unidades.

||| Diseño físico de la base de datos

En este paso hay que tener en cuenta:

- Aplicaciones a realizar sobre los datos
- Características particulares del SGBD

Si desea saber más acerca de estos temas, puede consultar las siguientes páginas.

Link	Descripción
http://ict.udlap.mx/people/carlos/is341/bases03.html	En esta página web encontrará conceptos adicionales y ejemplos de cómo se transforma paso a paso un Diagrama Entidad Relación a un Modelo Relacional.
http://ict.udlap.mx/people/carlos/is341/bases07.html	En esta página web encontrará conceptos adicionales y ejemplos de cómo se transforma un modelo físico a partir de modelo lógico.
http://www.ibm.com/support/knowledgecenter/es/SSQ2R2_9.5.1/com.ibm.datatools.core.ui.doc/topics/cphysmod.html	En esta página web encontrará conceptos básicos de modelos de datos físicos.
http://www.ceidis.ula.ve/cursos/pqcom/p/ing_data/dis.html	Página web de la Ingeniería de Datos y los Sistemas de Bases de Datos.
http://repositori.uji.es/xmlui/bitstream/handle/10234/24183/s18.pdf?sequenc e=6&isAllowed=y	Página web de Diseño de base de datos

*“Tus padres y tus maestros son tus fieles y desinteresados amigos;
apréchálos que no podrán acompañarte por mucho tiempo”*

Cicerón