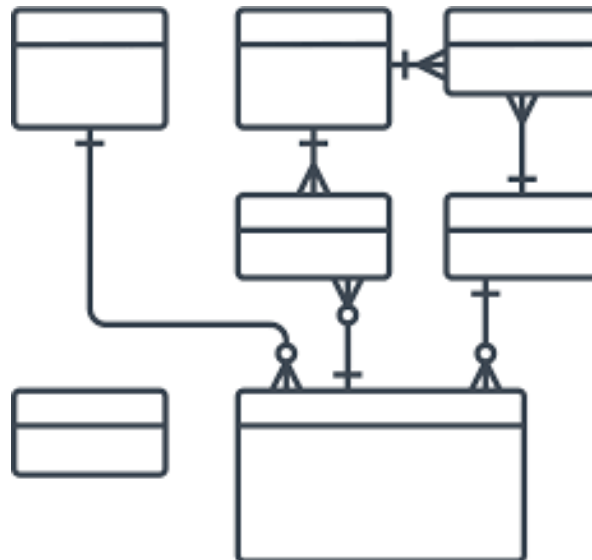


**UNIVERSIDAD DE LAS REGIONES AUTÓNOMA DE LA
COSTA CARIBE NICARAGÜENSE
URACCAN**

Recinto Bilwi

MODULO

Base de Datos I



CARRERA

Ingeniería en Sistemas

AUTOR

Dicxie Danuard Madrigal Brack

Bilwi, Puerto Cabezas 14 de septiembre del 2018

Índice

I.	Presentación de las unidades	1
II.	Introducción	2
III.	Objetivos generales.....	3
IV.	Metodología	¡Error! Marcador no definido.
V.	Sistema de evaluación.....	5
VI.	DESARROLLO.....	6
	Unidad I: Introducción a las bases de datos	6
1.1.	Evolución de las bases de datos.....	6
1.2.	Definición y objetivos de base de datos.....	8
1.3.	Tipos de Bases de datos. Modelos de base de datos.....	8
1.4.	Usos y aplicaciones de las bases de datos.....	10
1.5.	Arquitectura de base de datos	11
1.5.1.	Niveles de abstracción de una base de datos	11
1.5.2.	Independencia lógica y física de los datos	12
1.6.	Estructura general de un sistema de base de datos	13
1.6.1.	El sistema gestor de base de datos (SGBD).....	13
1.6.2.	Los usuarios de la base de datos.....	13
1.6.3.	El administrador de la base de datos	14
	UNIDAD II: MODELADO DE DATOS	17
2.1.	Modelo de datos.....	17
2.1.1.	Definición de modelo de datos	17
2.2.	Modelo entidad relación.....	18
2.2.1.	Entidades.....	18
2.2.2.	Atributos.....	19
2.2.2.1.	Atributos compuestos frente a atributos simples	19
2.2.2.2.	Atributos Mono-valor y Multi-valor.....	19
2.2.2.3.	Atributos almacenados y derivados	19
2.2.2.4.	Atributos clave	20
2.2.3.	Cardinalidad de las entidades	20
2.3.	Modelo Relacional	24
2.4.	Estructura del modelo relacional (atributo, dominio, tuplas)	24
2.5.	Definición de relación	25
2.6.	Propiedades de una relación	25
2.7.	Claves relacionales.....	25
	Unidad III: Diseño de bases de datos relacionales	28
3.1.	Anomalías de inserción, eliminación, y actualización de datos.	28
3.2.	Dependencias funcionales.....	29
	Unidad IV: Lenguaje de Base de Datos.....	30
VII.	Lista de referencias	31
VIII.	Glosario.....	32
IX.	Anexo.....	33

I. Presentación de las unidades

El presente módulo de Base de Datos I está compuesto de cuatro unidades:

Unidad I Introducción a las bases de datos: Dentro de esta unidad se define conceptos básicos, se describe la evolución, características así mismo la arquitectura que está compuesto la base de datos

Unidad II Modelo de datos: en esta unida estudia el modelo Entidad-Relacional y el Modelo Relacional como una forma de representar las informaciones almacenados en las bases de datos

Unidad III Diseño de base de datos relacionales: es está unidad se centra en los aspectos de diseño de las bases de datos relacionales, que son las normalizaciones, la integridad de referencias, seguridad, usuario, roles, vista y los diccionarios de datos.

Unidad IV Lenguaje de base de datos: en esta unidad se estudia de manera general el lenguaje de base de datos SQL, lenguaje que permite definir las estructuras de una base de datos relacionales implementando integridad referencial.

II. Introducción

En este módulo de Base de Datos I tiene como objetivo que los estudiantes se apropien del conocimiento y las tecnologías elementales de las bases de datos. Debido a que las mayorías de las bases de datos es encuentran en sistemas computacionales, y como futuro profesional de la costa caribe de Nicaragua debe ser capaz de diseñar, crear y administrar bases de datos.

Debido lo mencionado, este módulo está organizado en 4 unidades: Introducción a la base de datos, Modelos de datos, Diseño de base de datos relacionales y lenguaje de base de datos.

Al finalizar el modulo los estudiantes debe de haber desarrollado las habilidades para el diseño de las bases de datos, usando los modelos e integrando los conocimientos adquiridos, la estructura de la base de datos, la seguridad de la base de datos y la actividad de los datos. A través de un proyecto final de curso donde se diseña una base de datos que satisfaga la necesidad de información de algún negocio o empresa.

III. Objetivos generales

- Conoce los conceptos básicos, principales ventajas, evolución histórica de la base de datos y su uso en la implementación para los sistemas de información
- Definir y diferenciar los modelos de datos y sus componentes
- Explicar los conceptos básicos y fundamentos de los sistemas de bases de datos.
- Conocer las características, funcionalidades y estructura de las bases de datos.
- Conocer las herramientas necesarias para el diseño e implementación de bases de datos.
- Comprende el modelado de datos y los procesos de normalización.
- Conoce los lenguajes de bases de datos para su implementación

IV. Orientación metodología

El módulo está dividido en 4 unidades, al finalizar cada unidad los estudiantes presentarán un avance de su proyecto final de curso. Así mismo en la

V. Sistema de evaluación

VI. Desarrollo

Unidad I: Introducción a las bases de datos

Contenido

- 1.1. *Evolución de las bases de datos*
- 1.2. *Definición y objetivo de la base de datos*
- 1.3. *Tipos de bases de datos. Modelos de bases de datos*
- 1.4. *Uso y aplicaciones de las bases de datos*
- 1.5. *Arquitectura de base de datos*
 - 1.5.1. *Niveles de abstracción de una base de datos*
 - 1.5.2. *Independencia física y lógica de los datos*
- 1.6. *Estructura general de un sistema gestor de base de datos*
 - 1.6.1. *El sistema gestor de base de datos*
 - 1.6.2. *Los usuarios de la base de datos*
 - 1.6.3. *El administrador de la base de datos*

Objetivos

- Conoce los conceptos básicos de la base de datos y su evolución histórica.
- Describir los diferentes modelos de base de datos
- Explicar la arquitectura de una base de datos
- Describir los diferentes actores que utilizan las bases de datos

1.1. Evolución de las bases de datos.

Los orígenes de las bases de datos se remontan a la Antigüedad donde ya existían bibliotecas y todas clases de registro. Además, se utilizaba para recoger información sobre las cosechas y censos. Sin embargo, su búsqueda era lenta y poco eficaz y no se contaba con la ayuda de máquinas que pudiese reemplazarlos en el trabajo manual.

Posteriormente, el uso de las bases de datos se desarrolló a partir de las necesidades de procesamiento de datos y la gestionar la información. La tecnología moderna de bases de datos, ha tenido lugar desde la tecnología de acceso a datos que se desarrolló como una de los primeros métodos primitivos de los años cincuenta. Los proyectos de exploración espacial han contribuido mucho en los descubrimientos en la industria de la ciencia y la tecnología.

Dentro del proyecto lunar Apolo la North American Aviation (NAA) creó un sistema jerárquico de archivos llamado Método de acceso generalizado a actualizaciones (GUAM, Generalized Update Access Method) en 1964. Posteriormente la empresa IBM se unió a la NAA para desarrollar GUAM como la primera base datos del modelo relacional disponibles para la venta; la cual se le llamó Information Management System (IMS) y la distribuyo en 1966. Durante esta misma época, General Electric creó de manera interna la primera base de datos en el modelo de red, bajo la

dirección del destacado científico computacional Chales W. Bachman y la nombro Integrated Data Store (IDS).

En 1967, la Conference of Data Systems Languages (CODASYL) un grupo industrial, formó el Database Task Group (DBTG) y comenzó a trabajar en un conjunto de normas para el modelo de red. Como respuesta a las críticas de la restricción de “elemento principal único” del modelo jerárquico, IBM introdujo una versión de IMS que eliminaba el problema al permitir que los registros tuvieran un elemento principal “físico” y varios “lógicos”.

En junio de 1970, E. F. (Ted) Codd, un investigador de IBM (y luego socio de la empresa), publicó un documento de investigación titulado “*A relational model of data for large shared data banks*” (Un modelo relacional de datos para grandes bancos de datos compartidos) en Communications of the ACM, la publicación periódica de la Association for Computing Machinery, Inc. (La publicación se obtiene fácilmente en Internet.) En 1971, el DBTG de CODASYL publicó sus normas, que tardaron más de tres años en elaborar. Esto inició cinco años de debate acalorado acerca de cuál modelo era el mejor.

Dentro de las afirmaciones del DBTG de CODASYL era lo siguiente:

- El modelo relacional era demasiado matemático.
- No era posible crear de manera eficiente un modelo relacional
- Las aplicaciones necesitan procesar la información de registro a registro

Los promotores del modelo relacional realizaron los siguientes argumentos:

- La administración de los datos no era tan complicada como lo propuesto del DBGT.
- Era difícil realizar consulta orientado a conjunto en el lenguaje DBTG
- La falta de sustento formal de la matemática en el modelo de red.

Debido a las afirmaciones presentado por las partes sobre los modelos estaban en los cierto, sin embargo, durante esa época el interés por el modelo red se desvaneció de la década de los 1970, con la evolución de las tecnologías de las bases de datos computacionales demostró que el modelo relacional era la mejor opción debido a:

- Surgieron lenguajes estructurado de consulta SQL que no era tan matemático
- Una eficiencia razonable a comparación con el modelo de red
- Incorpora el leguaje SQL para procesar un conjunto de datos mediante un método de un registro a la vez
- El descubrimiento de una nueva disciplina, conocida como calculo relacional

1.2. Definición y objetivos de base de datos.

Una base de datos es una colección de datos auto-descriptible, interrelacionados y administrados como unidad, que describe un modelo. Es importante entender plenamente esta definición, para ellos de explica cada uno de los términos importante.

- **Datos:** Con la palabra datos nos referimos a hechos (datos) conocidos que se puede grabar y que tiene un significado implicado implícito.
- **Una base de datos es auto-descriptivo:** Contiene, además de los datos de los usuarios, unas descripción de su propia estructura. Esta descripción se le llama **diccionario de datos** (o también **directorio de datos** o **metadatos**).
- **Una base de datos es un conjunto de registro interrelacionado:** las bases de datos están por campos; los integran registro, y los registros se agrupan en tabla. Además, incluye que se utiliza para representar la relación entre datos, y también para mejorar el desarrollo de las aplicaciones de una base de datos
- Una base de datos es modelo: es decir, que es un modelo de la realidad, o de alguna parte de la realidad en la que se relaciona con el negocio. sin embargo, este no es verdad. Una base de datos no es un modelo de la realidad ni una parte de éste, sino un modelo del modelo de usuario.

1.3. Tipos de Bases de datos. Modelos de base de datos.

La información de una base de datos se describe a través de los modelos. Los modelos representan la forma en que está organizado los datos dentro de la base de datos. Los modelos son:

1.3.1. Modelo jerárquico

Las primeras bases de datos seguían el modelo jerárquico, que evoluciono a partir de los sistemas de archivo que reemplaza las bases de datos, con los registros ordenamos en una jerarquía similar a un organigrama. Cada archivo de un sistema de archivo se convirtió en un tipo de registro, o nodo.

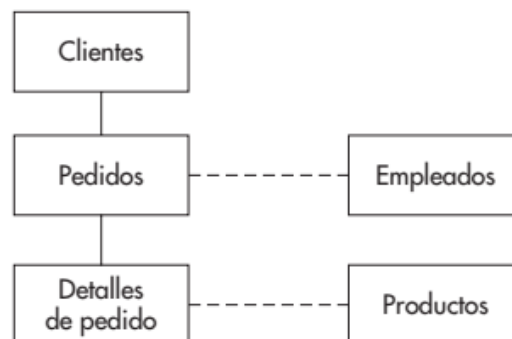


Figura 1: Estructura de un modelo jerárquico

1.3.2. Modelos de red

El modelo de red evolución del casi al mismo tiempo que el modelo jerárquico. es debido que al igual que el modelo jerárquico los tipos de registro (o sencillamente registro) representa lo que sería archivos separados en un sistema de archivos simples, y esos registros se enlaza mediante relaciones propiedad-miembro o conjuntos, en la terminología del modelo de red.

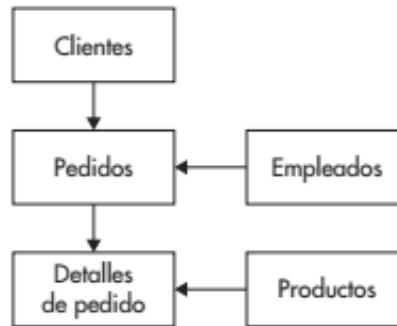


Figura 2: Estructura de un modelo de Red

1.3.3. Modelo de relacional

Este modelo representa la información en forma de tabla bidimensionales, como lo hace una hoja de cálculo.

Tabla Clientes						
Id de cliente	Compañía	Apellidos	Nombre	Cargo	Ciudad	Estado o provincia
6	Compañía F	Pérez-Olaeta	Francisco	Jefe de compras	Milwaukee	WI
26	Compañía Z	Pinto	Armando	Ayudante de contabilidad	Miami	FL

Tabla Pedidos					
Id de pedido	Id de cliente	Id de empleado	Fecha de pedido	Fecha de envío	Gastos de envío
51	26	9	05/04/2006	05/04/2006	\$60.00
56	6	2	03/04/2006	03/04/2006	\$0.00
79	6	2	23/06/2006	23/06/2006	\$0.00

Tabla Empleados			
Id	Apellidos	Nombre	Cargo
2	Escolar	Jesús	Vicepresidente de ventas
5	San Juan	Patricia	Jefe de ventas
9	Chaves	Francisco	Representante de ventas

Figura 3: Representación de un modelo relaciona

1.3.4. Modelo Orientado a Objeto

La idea de base de datos Orientados a Objetos (OO) tuvo su inicio en 1970 pero no fue hasta en el año 1990 que este tuvo uso comercial. Este tipo de bases de datos surgió alta la limitación de las modelos relacionales para guardar datos más complejos, como imágenes, dibujos, audio y videos.

Los objetos son grupos lógicos de datos relacionados y de lógica de programa que representa algo de la real, como un cliente, empleado, pedido o producto.

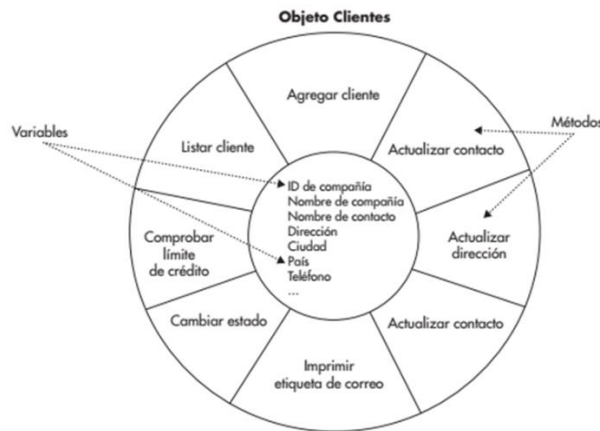


Figura 4: Estructura de un modelo orientado a objeto

Modelo relacional Orientado a Objeto

Este modelo es la combinación de los orientado a objeto y el modelo relacional, donde, se aporta al modelo relacional algunos de los beneficios del modelo OO, una de esto es la encapsulación de que disminuyo las modificaciones del sistema, además que permite utilizar tipos de datos complejos en el modelo relacional.

1.4. Usos y aplicaciones de las bases de datos.

El propósito de las bases de datos es ayudar a las personas a dar seguimiento a las cosas. las aplicaciones clásicas de base de datos tales como órdenes, clientes, empleados, llamadas telefónicas. Con la llegada de la Internet se ha utilizados las bases de datos para venta de productos en líneas. a continuación, se presenta un ejemplo.

Pintora de casa Luisa Gutiérrez

Luisa Gutiérrez es una pintora profesional de casa que es la propietaria y opera de pequeña compañía “Colores Vivos” en la ciudad de Bilwi, integrada por ella, otro pintor profesional y, como es necesario pintores de medio tiempo. Luisa ha estado en el negocio de durante diez años y se ha ganado la reputación de ser profesional y ofrece precios razonables. Ella consigue la mayoría de los trabajos con clientes que la contrataron en varias ocasiones y también a través de la referencias y recomendaciones que dan sus clientes a otras personas; además consigue algo de trabajos con los constructores y los diseñadores profesionales de interiores.

Los clientes recuerdan mejor a Luisa de lo que los recuerda ella los recuerda a ellos. De hecho, algunas veces se avergüenza cuando un cliente le llama y le dice algo como “Hola, Luisa habla Jhon Maples. Pintaste mi casa hace 3 años”. Ella sabe que debería de recordar a todos aquel que le llaman para el trabajo realizado., pero más

de 50 casa al año y es difícil de recordar algunos clientes en particular. Esta situación empeora ellos le dice: “A mi vecino le gusto trabajo que realizaste en nuestra casa y le gustaría algo similar para la suya”.

Con el fin de ayudar a su memoria y dar seguimiento a sus registros. Luisa tenía un especialista que desarrolló una base de datos y las aplicaciones que utiliza en su computadora personal. La base de datos almacena registro de los clientes, trabajos y fuente de referencias

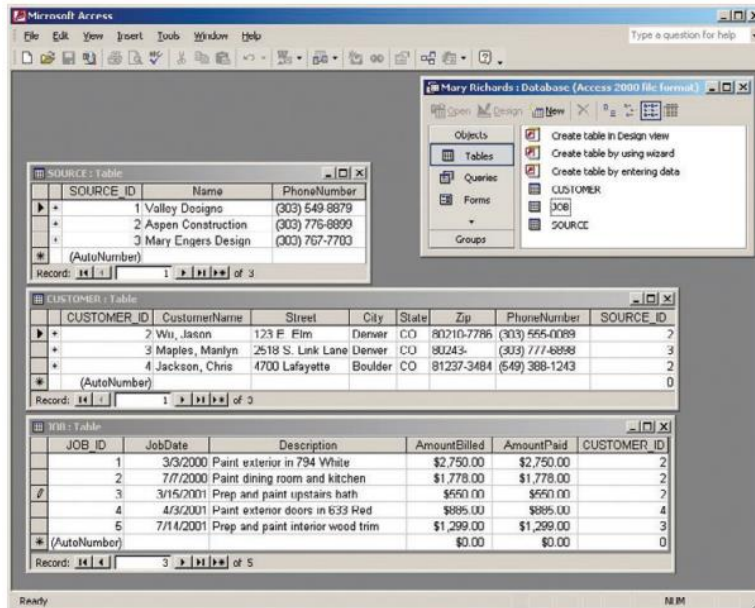


Figura 5: Base de datos de Luisa Gutierrez

1.5. Arquitectura de base de datos

En el año 1975, el comité ANSI-SPARC (American National Standard Institute – standard planning and Requirements Committee) propuso una arquitectura de tres niveles o capas para los sistemas de base de datos, que resultó muy útil a la hora de conseguir estas tres características: el nivel interior física o de máquina, el nivel exterior o de usuario, el nivel conceptual. Así mismo describe las interacciones entre estos tres niveles y todos los elementos que conforma uno de ellos.

1.5.1. Niveles de abstracción de una base de datos

La base de datos puede tener tres niveles de abstracción. Los niveles de abstracción forman la arquitectura de tres niveles de capa y se describen mediante tres esquemas que son descriptores de su estructura. El propósito de arquitectura de tres niveles es separar el modelo de usuario de la estructura física de la base de datos.

El **nivel interno** tiene un esquema interno, que describe la estructura de almacenamiento físico de la base de datos. El esquema interno utiliza un modelo de

datos físico y describe todos los detalles del almacenamiento de datos y las rutas de acceso a la base de datos

El **nivel conceptual** tiene un esquema conceptual, que describe la estructura de toda la base de datos para una comunidad de usuarios. El esquema conceptual oculta los detalles de las estructuras de almacenamiento físico y se concentra en describir las entidades, los tipos de datos, las relaciones, las operaciones de los usuarios y las restricciones. Normalmente, el esquema conceptual se describe con un modelo de datos representativo cuando se implementa un sistema de bases de datos. Este esquema conceptual de implementando se basa a menudo en un diseño de esquema conceptual en un modelo de datos de alto nivel.

El **nivel de vista** o externo incluye una cierta cantidad de **esquemas externos** o **vistas de usuario**. Un esquema externo describe la parte de la base de datos en la que un grupo de usuarios en particular está interesado y le oculta el resto de la base de datos. Como en el caso anterior, cada esquema externo se implementa normalmente mediante un modelo de datos representativo, posiblemente rasado en un diseño de esquema externo de un modelo de datos de alto nivel.

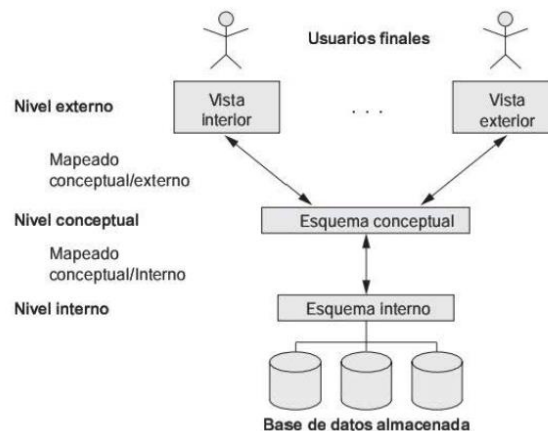


Figura 6: Niveles de abstracción de una base de datos

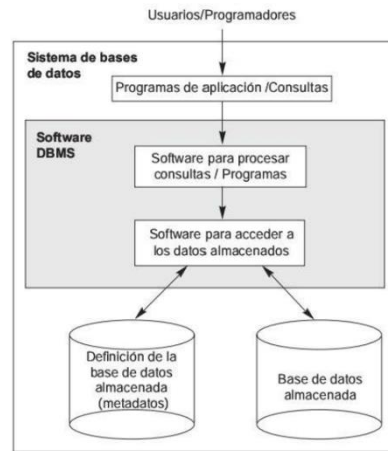
1.5.2. Independencia lógica y física de los datos

La arquitectura de tres esquemas se puede utilizar para explicar el concepto de independencia de los datos, que puede definirse como la capacidad de cambiar el esquema en un nivel de un sistema de bases de datos sin tener que cambiar el esquema en el siguiente nivel más alto. Se pueden definir dos tipos de independencia de datos:

Independencia lógica de datos. Es la capacidad de cambiar el esquema conceptual sin tener que cambiar los esquemas externos o los programas de aplicación. Es posible cambiar el esquema conceptual para expandir la base de datos (añadiendo un tipo de registro o un elemento de datos), para cambiar las restricciones o para reducir la base de datos (eliminando un tipo de registro o un elemento de datos).

Independencia física de datos. Es la capacidad de cambiar el esquema interno sin que haya que cambiar el esquema conceptual. Por lo tanto, es necesario cambiar los esquemas externos

1.6. Estructura general de un sistema de base de datos



1.6.1. El sistema gestor de base de datos (SGBD).

Es una colección de programas que permite a los usuarios crear y mantener una base de datos. El SGBD es un sistema de software el propósito general que facilita los procesos de definición, construcción, manipulación y compartición de bases de datos entre varios usuarios y aplicaciones

1.6.2. Los usuarios de la base de datos

- Los **usuarios finales** son las personas cuyos trabajos requieren acceso a la base de datos para realizar consultas, actualizaciones e informes; la base de datos existe principalmente para ser utilizada. Los usuarios finales se pueden clasificar en varias categorías:
 - Los **usuarios finales** casuales: acceden ocasionalmente a la base de datos, pero pueden necesitar una información diferente en cada momento.
 - Los **usuarios finales principales** o **paramétricos**: constituyen una parte considerable de los usuarios finales de las bases de datos. Su labor principal gira entorno a la consulta y actualización constantes de la base de datos, utilizando tipos de consultas y actualizaciones estándar que se han programado y probado cuidadosamente
 - Los **usuarios sofisticados** se encuentran los ingenieros, los científicos, los analistas comerciales y otros muchos que están completamente familiarizados con el DBMS a fin de implementar sus aplicaciones y satisfacer sus complejos requisitos.
 - Los **usuarios finales independiente** mantiene bases de datos personales utilizando paquetes de programas confeccionados que

proporcionan unas interfaces fáciles de usar y basadas en menús o gráficos.

1.6.3. El administrador de la base de datos

Los administradores de base de datos (DBA, Database administrator) son los encargados del acceso autorizado a la base de datos, de la coordinación y monitoreo del uso, y de adquirir los recursos software y hardware necesarios. Además de las funciones mencionada anteriormente el DBA.

Investigue

- a) Las *características y propiedades* de una base de una base de datos
- b) Investigue ventajas y desventajas de los *sistemas basados en archivos y los sistemas de gestor de base de datos*
- c) Las personas (actores) que hacen de las bases de datos

Ejercicios prácticos

Compañía FiredUp

1. La compañía FiredUp es un negocio pequeño, cuyos propietarios son Curt y Julia Robard. La sede está en Brisbane, Australia. FiredUp fabrica y vende una estufa para campo ligera, llamada FiredNow. Curt, quien trabajó previamente como ingeniero aéreo parcial, inventó y patentó un mechero con el que la estufa permanecerá encendida aun cuando el viento sea superior a los 160 kilómetros por hora. Julia, una diseñadora industrial muy capaz, desarrolló un elegante diseño plegadizo pequeño, ligero, fácil de instalar y muy estable. Los Robard fabrican las estufas en su cochera y las venden directamente a sus clientes a través de Internet, fax o correo.

Los propietarios de FiredUp necesitan dar seguimiento a las estufas que han vendido, con el fin de poder entrar en contacto con sus usuarios en caso de fallas del producto u otros asuntos relacionados con éste. También piensan que podrían usar su lista de clientes para comercializar otros productos que desarrollen en el futuro.

Empresa Willson House

Preguntas de comprobación

1. *¿Defina las siguientes palabras: Datos, Base de datos, Sistemas de gestor de bases de datos?*
2. *Explique las categorías de modelo de datos*

3. *Describe la arquitectura de tres niveles de una base de datos*
4. *¿Cuáles son las diferencias entre Administrador de datos(DA), y un Administrador de base de datos(DBA)?*
5. *¿cuáles son las responsabilidades de Administrador de base de datos y los diseñadores de bases de dato?*
6. *En relación con el modelo jerárquico de bases de datos, ¿cuál de las siguientes afirmaciones es verdadera?*
 - a. Fue desarrollado primero por Peter Chen.
 - b. Los datos y los métodos se guardan juntos en la base de datos.
 - c. Cada nodo puede tener muchos padres.
 - d. Los registros se conectan mediante apuntadores a una dirección física.
7. ***En relación con el modelo relacional de bases de datos, ¿cuál de las siguientes afirmaciones es verdadera?***
 - a. Fue desarrollado por Charles Bachman
 - b. Los datos y los métodos se guardan juntos en la base de datos.
 - c. Los registros se conectan mediante apuntadores a una dirección física.
 - d. Los registros se conectan mediante elementos de datos comunes en cada registro.
8. ***En relación con el modelo orientado a objetos de bases de datos, ¿cuál de las siguientes afirmaciones es verdadera?***
 - a. Fue desarrollado por Charles Bachman.
 - b. Los datos y los métodos se guardan juntos en la base de datos.
 - c. Los datos se presentan en tablas bidimensionales.
 - d. Los registros se conectan mediante elementos de datos comunes en cada registro.

Instrucciones del proyecto

En esta sección describe un proyecto. Deberá de estudiar el proyecto de muestra y utilizarlo como modelo para llevar acabo los pasos para su proyecto. Si puede hacer entrevista a personas que estén familiarizadas con el entorno descrito en el proyecto. Con base en sus entrevistas, la descripción escrita y su propio análisis del proyecto que eligió, realizar lo siguiente:

- Definir el tema de investigación
- Realizar una descripción general del proyecto
- Operaciones básicas
- Describir la necesidad de información
- Escriba el formato de cada documento de entrada que proporcione información a almacenar en la base de datos
- Escriba el formato de cada reporte de rutina a producir usando la base de datos.

Unidad II: Modelado de datos

Contenidos

- 2.1. *Modelo de datos*
 - 2.1.1. *Definición de modelo de datos.*
- 2.2. *Modelo entidad relación.*
 - 2.2.1. *Entidades*
 - 2.2.2. *Atributos*
 - 2.2.3. *Atributos compuestos frente a atributos simples*
 - 2.2.4. *Atributos mono-valor y multi-valor*
 - 2.2.5. *Atributos almacenados y derivados*
 - 2.2.6. *Atributos clave*
 - 2.2.7. *Cardinalidad de las entidades*
- 2.3. *Modelo Relacional*
- 2.4. *Definición de relación*
- 2.5. *Estructura del modelo relacional*
- 2.6. *Propiedades de una relación*

Objetivos

- Conocer los diferentes modelos de datos y sus componentes
- Conocer las características, funcionalidades y estructura de las bases de datos.

2.1. Modelo de datos

2.1.1. Definición de modelo de datos

El modelo representa de un objeto y sucesos del “*mundo real*”, así como de sus asociaciones. se trata de una abstracción que se Concreta el aspecto esencial e inherentes de una organización enumera las propiedades accesorias. Un modelo de datos representa a la propia organización y a sus usuarios finales comunica de forma precisa. La comprensión de los datos de la organización. podemos considerar que los modelos de datos comprenden tres componentes.

- Una **parte estructural**, compuesto por un conjunto de reglas que son las que deciden Cómo pueden Construir la base de datos.
- Una **parte manipulativa**, Qué es y los tipos de operaciones que puede realizar sobre los datos (las operaciones empleadas para actualizarlo extraer datos de una base de datos Y para poder modificar la estructura física de la base datos).
- Posiblemente, un conjunto de **restricciones de integridad**, que Precisión de los datos.

El propósito de un modelo de datos representar los datos Y hacer que sean comprensibles. Si lo consigue, se lo podrá utilizar con facilidad para diseñar una base de datos. para empezar arquitectura ANSI- SPARC.

2.2. Modelo entidad relación.

El modelo Entidad-Relación es el modelo de datos más ampliamente usado para el diseño de bases de datos. Fue creado por *Peter Chen* en el año 1976, en 1988 el ANSI seleccionó el modelo ER como el modelo estándar para los sistemas de diccionarios de recursos de información.

Formalmente, los diagramas E-R son un lenguaje gráfico para describir conceptos. Informalmente, son simples dibujos o gráficos que describen la información que trata un sistema de información y el software que lo automatiza

El modelo de Entidad-Relación(ER) es un modelo de datos basado en objeto que utiliza conceptos tales como *entidad*, *atributos* y *relaciones*. El modelo En modelo ER se ha considerado como una de las técnicas principales para el diseño de base de datos.

2.2.1. Entidades

Es una cosa del mundo real con una existencia independiente. Una entidad puede ser un objeto con una existencia independiente. Un objeto puede ser un objeto con una existencia real (Persona, coche, casa o empleados) o puede ser un objeto con una existencia conceptual (por ejemplo, una empresa, un trabajo, un curso o una universidad)



Figura 7: Representación de una entidad

Las entidades se pueden clasificar en entidades fuertes y débiles

- Entidades fuertes: una entidad cuya existencia no depende de ninguna otra entidad
- Entidades débiles: son entidades cuya existencia depende de alguna otra entidad

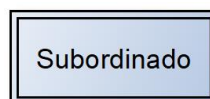


Figura 8: Representación de un entidad débil

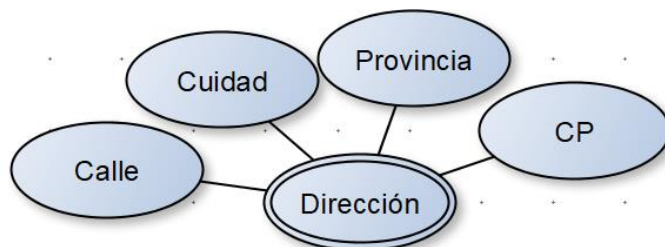
2.2.2. Atributos

Los atributos son las particularidades que describen a una entidad. Por ejemplo, una entidad EMPLEADO se puede mediante nombre, la edad, la dirección, el sueldo y el trabajo que desempeña



2.2.2.1. Atributos compuestos frente a atributos simples

En el modelo ER se tiene varios tipos de atributos: **simples** y los **atributos compuestos**. Los **atributos compuestos** dividir en subpartes más pequeñas que representan atributos más básicos con significados independientes, por ejemplo, el atributo *dirección* de un EMPLEADO se puede dividir en *calle*, *cuidad*, *provincia* y *código postal*. Se le llama atributos **simples** o **atómicos** a todos aquellos atributos que no son divisibles. Los atributos compuestos son unión de varios atributos simples



2.2.2.2. Atributos Mono-valor y Multi-valor

La mayoría de los atributos solo tiene un valor para cada entidad en particular; de otra forma a estos tipos de atributo reciben el nombre de monovalor o de un solo valor. Por ejemplo, la edad es un atributo monovalor de una persona. En algunos casos, un atributo puede tener conjunto de valores para la misma entidad (por ejemplo, un atributo Colores para un coche, un atributo Licenciaturas para una persona).

2.2.2.3. Atributos almacenados y derivados

En algunos casos, dos (o más) valores de atributos están relacionada (por ejemplo, los atributos Edad y Fecha de Nacimientos de una persona). Para una entidad de la persona en particular el valor de *edad* puede determinarse a partir de la fecha actual (el día de hoy) y el valor de la fecha de nacimiento. El atributo edad se denomina **atributos derivado** y se dice que se ha derivado del atributo fecha de nacimientos, que es denominado **atributos almacenados**. Algunos valores de los atributos se derivan de entidades relacionadas, por ejemplo, un atributo número empleado de una entidad Departamentos puede derivarse contando el número de empleado relacionado.

Figura 9: Representación de un atributo simple
Figura 10: Representación de atributo compuesto

Valores null (nulo) en algunos casos, es posible que una entidad en particular no tengo un valor aplicable para un atributo. Por ejemplo, el número de apartamento, solo es aplicable a las direcciones correspondiente a edificios de apartamento, y no a otros tipos de residencia

2.2.2.4. Atributos clave

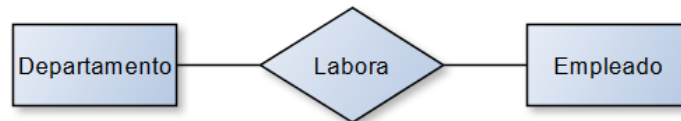
Una de las restricciones importante de las entidades de una de entidad es la **clave** o **restricción de unicidad** de un atributo. Una entidad normalmente tiene un atributo cuyo valor son distintos para cada entidad individual.



Figura 11: Representación de un atributo clave

2.2.3. Cardinalidad de las entidades

Relaciones: una relación es conjunto de asociaciones significativas entre entidades. Cada relación recibe un nombre que describe su función. Un ejemplo de relación sería *labora* que asocia a una entidad *Empleado* y un *Departamento*.



Multiplicidad o cardinalidad: es el número (o rango) de posibles instancias de una entidad que puede relacionarse con una única instancia de otra entidad a través de una relación concreta. En otras palabras, restringe la forma en que las entidades se relacionan. Se trata de una representación de políticas (Reglas de negocio) establecidas por el usuario o la empresa ejemplo:

- Un empleado gestiona una sucursal (1:1)
- Un empleado controla inmuebles en alquiler (1:*)
- Los periódicos anuncian inmuebles en alquiler (*:*)

Relación uno a uno (1:1)

Es una asociación en que una instancia de una entidad **A** se puede asociar cuando mucho a una sola instancia de una entidad **B**, y viceversa. Tal y como aparece en la figura 14. Un ejemplo de esta relación sería la relación empleado y sucursal, donde un empleado gestiona solamente una sucursal y la sucursal solo es administrado por un solo empleado, tal y como se ve en la figura 15.

Figura 12: Ejemplo de una relación

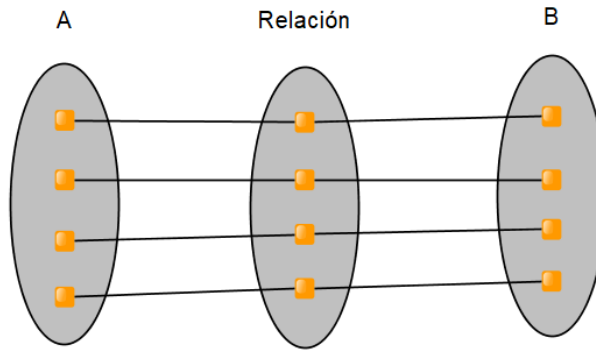


Figura 13: relación 1:1



Figura 145: Relación de una a uno entre empleado y departamento

Relación uno a varios1:*

Es una asociación en que una instancia de una entidad **A** se puede asociar a mucha instancia de una entidad **B**, pero la entidad **B** solo estará asociada únicamente a una entidad **A**, como se presenta en la figura 16. Un ejemplo de una cardinalidad de este tipo sería la relación que existen entre Empleado y Departamentos, donde en un departamento laboran varios empleados, mientras que un empleado solo puede laborar en un solo departamento véase la figura 17.

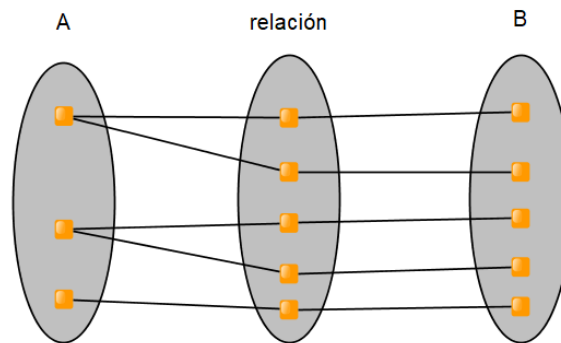


Figura 15: Representación de una relación una a muchos

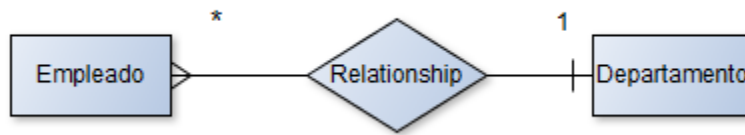


Figura 16: relación de uno a muchos entre la entidad Departamento y Empleado

Relación de muchos a muchos *:*

Es una asociación en que una instancia de una entidad **A** se puede asociar a mucha instancia de una entidad **B**, y una instancia de una entidad **B** se puede asociar varias o muchas veces a una instancia de una entidad **A**. por ejemplo una relación de este tipo se da con la entidad Periódico y la entidad Propiedad, donde un periódico puede realizar anuncio de varias propiedades, una propiedad puede ser anunciado en varios periódicos.

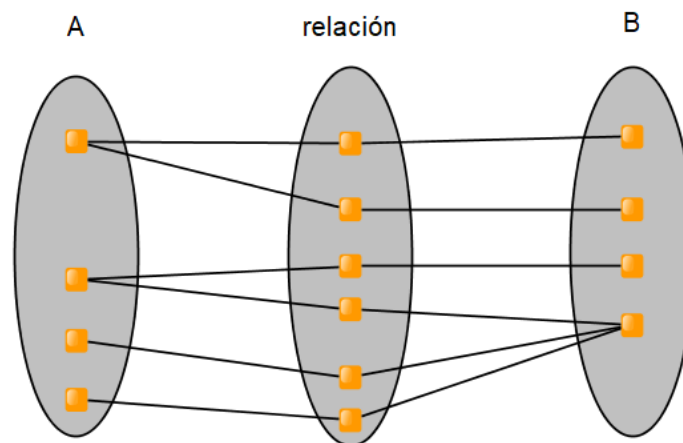


Figura 17: Representación de una relación mucho a mucho

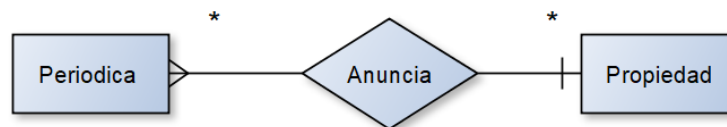


Figura 18: ejemplo de una relación de muchos a muchos entre Periódico y Propiedad

Ejemplo diagrama E-R de DreamHouse

DreamHouse es una empresa, dedicada en la venta y alquiler de inmuebles (los inmuebles puede tener propietario que no precisamente sea la misma) que tiene sucursales en las ciudades Managua, Estelí, Nueva Guinea, Bluefields y Bilwi. El genérate de la empresa desea administrar la información de sobre el alquiler de los

bienes inmuebles que realizan a sus clientes. Además, de administrar la información de sus empleados.

- La empresa tiene varias sucursales la información que se almacena de las sucursales son: un numero único para cada sucursal, calle, barrio y código postal.
- En cada sucursal laboran varios empleados la información registrado de los empleados en cada sucursal es: un código único de empleado, nombre, apellido, sexo, fecha en que inicio a laborar en la sucursal y salario
- Un inmueble es administrado por una sucursal y además la propiedad puede estas a cargo de un empleado. La información gestionada de las propiedades son un numero único que identifica a cada inmueble en específico, calle, ciudad, código postal, tipo de inmueble, cantidad de cuartas, precio de renta.
- Un propietario poder dueño de varios inmuebles, en donde, se registra los datos como código único, nombre, apellido, dirección y teléfono.
- La empresa desea llevar un seguimiento de todos sus clientes para ellos desea guardar en la base de datos los siguientes datos: un número único para el cliente, nombre, apellido, teléfono, preferencias del tipo de inmueble que prefiere arrendar, y la cantidad máximo que el cliente daría para una renta.
- Los clientes pueden realizar varia arrienda de propiedades (al momento de realizar la arrienda se define el lapso fecha de inicio y fecha fin en que estará arriendo la propiedad)
- Además, los clientes pueden realizar visitas a las propiedades que lo realizan en una fecha y además al momento de realizar la visita los clientes realizan un comentario sobre

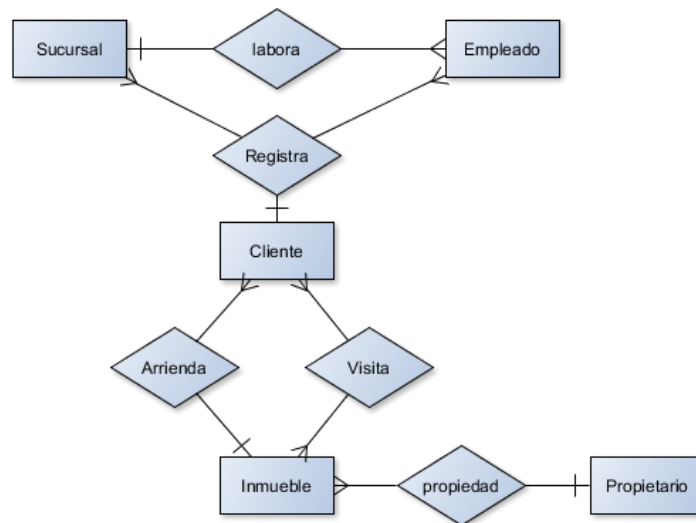


Figure 19: Diagrama E-R de la DreamHouse

2.3. Modelo Relacional

Como se mencionó anteriormente este modelo fue propuesto por E.F Codd en 1970. Dentro de los objetivos del modelo relacional eran:

- Permitir un alto grado de independencia de los datos. Los programas de aplicación no deben verse afectados por la modificación efectuada en la representación interna de los datos.
- Proporcionar una base teórica sólida que permite tratar con semántica de los datos y con los problemas de coherencia y redundancia. En particular en su artículo Codd introdujo los conceptos de **Normalización** tema que se estudiará en la próxima unidad.
- Permite la ampliación de lenguaje de manipulación de datos orientado a conjunto.

2.4. Estructura del modelo relacional (atributo, dominio, tuplas)

El **modelo relación** está basado en el concepto matemático de relación, la cual se representa físicamente en forma de **tabla**. Codd, que tenía formación matemática, utilizó terminologías sacadas de del campo de las matemáticas, principalmente de la *teoría de conjunto* y la *lógica de predicados*.

nomemp	nombre	apellido	cargo	sexo	inicio	salario	numdep
SL21	John	White	Administrador	M	1/10/1945	30000	B005
SG37	Ann	Beech	Asistente	F	10/11/1960	12000	B003
SG14	David	Ford	Superviso	M	24/3/1958	18000	B003
SA9	Mary	Howe	Asistente	F	19/2/1970	9000	B007
SG5	Susan	Brand	Administrador	F	3/6/1940	24000	B003
SL41	Julie	Lee	Asistente	F	13/7/1965	9000	B005

- **Atributos:** un atributo es una columna nominada de una relación. los atributos pueden aparecer en cualquier orden y la relación continua siendo exactamente igual, transmitiendo por lo tanto la información
- **Dominio:** un dominio es un conjunto permitidos de valores para uno o más atributos. Los dominios son característica extremadamente potente del modelo relacional. Cada atributo de una relación está definido sobre un dominio.
- **Tuplas:** una tupla es una fila de una relación., los elementos de una relación son las filas o tuplas de la tabla. Las tuplas puede aparecer en cualquier

orden y la relación continuará siendo la misma y transmitirá, por lo tanto, el mismo significado.

- **Grado:** un grado de una relación es el número de atributo que contiene.
- **Cardinalidad:** la cardinalidad de una relación es el número de tupla que contiene.

2.5. Definición de relación

Una relación es una tabla con columna y fila. En el modelo relacional, las relaciones se utilizan para almacenar información acerca de los objetos que hay que representar en la base de datos. Una relación se representa como

2.6. Propiedades de una relación

Una relación tiene las siguientes propiedades

- La relación tiene un nombre distinto de los demás nombres relaciones del esquema relacional.
- Cada celda de la relación contiene exactamente un valor atómico (único)
- Cada atributo tiene un nombre distinto
- Los valores de un atributo pertenecen todos al mismo dominio
- El orden de los atributos no tiene importancia
- El orden de las tuplas no tiene importancia, teóricamente (sin embargo, en la práctica, el orden puede afectar a la eficiencia de acceso a las tuplas)

2.7. Claves relacionales

En la relación no existe tupla duplicidad. Para poder identificar uno más atributos (denominados **claves relacionales**) que identifiquen de forma unívoca cada tupla de una relación.

- Superclave: un atributo o conjunto de atributos que identifica de forma unívoca cada tupla dentro de una relación
- Clave candidata: una superclave tal que ningún conjunto propio de la misma es una superclave de una relación.
- Clave principal: la clave candidata seleccionada para identificar las tuplas de forma unívoca dentro de la relación.
- Clave externa: un atributo, o conjunto de atributos, dentro de una relación que se corresponden de alguna relación.

Ejercicios

1. Pedidos por correo

Considere una base de datos PEDIDOS_CORREO en la que los empleados registran los pedidos de piezas por parte de los clientes. Realizar lo siguiente:

- Identifique las entidades, atributos, dominio y relaciones.
- Diseñe un diagrama E-R para la base de datos

Los requisitos en cuanto a datos son los siguientes:

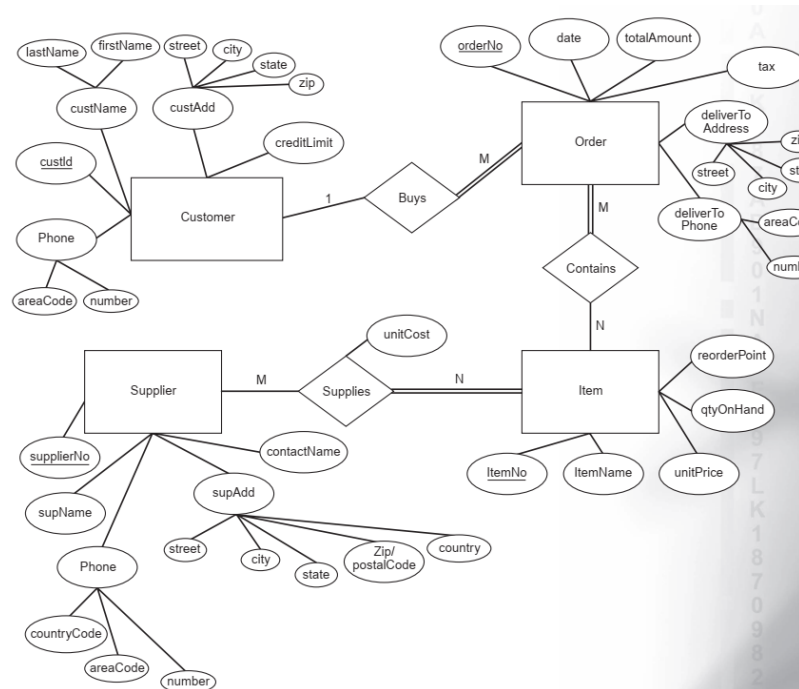
- La empresa de venta por correo tiene empleados identificados por un número de empleado único, además del nombre, los apellidos y el código postal.
- Cada cliente de la empresa está identificado mediante un número de cliente único, el nombre, los apellidos y el código postal.
- Cada pieza o repuesto vendido por la empresa está identificado por un número de repuesto único, un nombre, un precio y la cantidad en stock.
- Cada pedido efectuado por un cliente es registrado por un empleado y se le asigna un número de pedido que es único. Cada pedido contiene la cantidad especificada de uno o más repuestos, la fecha de recibo y la fecha de envío estimada. También se registra la fecha de envío real.

2. Cine

Considere una base de datos CINE en la que se registra información relativa a la industria cinematográfica. Los requisitos de datos se resumen a continuación:

- Cada película está identificada por su título y año de proyección. Además, la película tiene una duración en minutos, una productora y está clasificada según uno o más géneros (terror, acción, drama, etcétera). Cada película tiene uno o más directores, y uno o más actores, además de un resumen de la trama. Por último, cada película consta de ninguna o más citas reseñables por parte de los actores que aparecen en ella
- Los actores están identificados por su nombre y fecha de nacimiento, y aparecen en una o más películas. Cada actor tiene un papel en la película
- Los directores también están identificados por su nombre y fecha de nacimiento, y dirigen una o más películas. Es posible que un director también actúe en alguna película (incluyendo alguna que él o ella también haya dirigido).
- Las productoras están identificadas por su nombre y una dirección. Una productora produce una o más películas.

3. Crear el diagrama relacional a partir del ejercicio Pedidos por Correo
4. Crear el diagrama relacional a partir del ejercicio Cine
5. Elabore el diagrama relacional a partir del siguiente diagrama



Preguntas de comprobación

- Explique el papel del modelo de alto nivel en del diseño de base de datos
- Mencione y explique los conceptos básicos del modelo Entidad-Relacional
- ¿Cuándo se utiliza el concepto de entidad débil?
- Explique y de un ejemplo de Atributos compuesto, simples, mono-valor, multi-valor, almacenados, derivados y atributo clave?
- Mencione y explique los elementos de la estructura de una relación

Instrucciones del proyecto

- Realizar lista de entidades, atributos, y dominio.
- Diagrama el modelo E-R de la base de datos
- Diagramar el modelo relaciones de la base de datos

Unidad III: Diseño de bases de datos relacionales

Contenido

- 3.1. Anomalías de Inserción, eliminación y actualización de datos.
- 3.2. Dependencia funcional.
- 3.3. Normalización.
 - 3.3.1. 1FN, 2FN, 3FN y FNBC (forma normal de Boyce-Cood).
 - 3.3.2. 4FN y 5FN
- 3.4. Otros enfoques hacia el diseño de bases de datos
- 3.5. Integridad de bases de datos.
 - 3.5.1. Concepto.
 - 3.5.2. Restricciones básicas (NOT NULL, llave primaria)
 - 3.5.3. Integridad referencial.
- 3.6. Seguridad de bases de datos.
 - 3.6.1. Concepto de seguridad.
 - 3.6.2. Autenticación y autorización.
 - 3.6.3. Rol y privilegios de usuarios.
 - 3.6.4. Vistas y seguridad.
- 3.7. Diccionario de datos.
 - 3.7.1. Concepto.
 - 3.7.2. Contenido y función.

Objetivo

- Describir los problemas que presenta una relación no normalizada
- Diseñar relaciones utilizando la normalización y otros enfoques
- Explicar la importancia de la integridad de la base de datos, la seguridad y el diccionario de datos al momento de diseñar una base de datos

3.1. Anomalías de inserción, eliminación, y actualización de datos.

En una relación que no ha sido sometidos a un **proceso de normalización** presentas por lo general anomalías de inserción, eliminación actuación, redundancia e inconsistencia de los datos. A continuación, se presenta una relación con las anomalías, y que servirá de ejemplo para la explicación.

NomEmp	Nombre	Cargo	salario	codSucursal	sdireccion
SL21	John White	Administrado	30000	B005	22 Deer Rd, London
SG37	Ann Beech	Asistente	12000	B003	163 Main St, Glasgow
SG14	Elena Miranda	Supervisor	18000	B003	163 Main St, Glasgow
SA9	Sofia Sambran	Asistente	9000	B007	16 Argyll St, Aberdeen
SG5	Susa Brand	Administrado	24000	B003	163 Main St, Glasgow
SL41	Julia Lee	Asistente	9000	B005	22 Deer Rd, London

Figure 20: relación EmpleadoSucursal

Anomalía inserción

La hablar sobre *anomalías de inserción* se refiriera la situación en la que no es posible insertar un nueva tupla debido a que existe una dependencia de los valores de otra entidad. Estos tipos de anomalías surge cuando existen en una misma relación dos entidades. Existen dos tipos de anomalías de inserción, se utilizara la relación *EmpleadoSucursal* mostrado en la figura 20:

- Para insertar los detalles de un nuevo empleado en la relación ***EmpleadoSucursal*** debemos de incluir los detalles de la sucursal en el que trabaja el empleado. Por ejemplo, para insertar la información de un nuevo empleado de la sucursal B007, debemos incluir los detalles correctos de la sucursal B007.
- Para insertar los detalles de una nueva sucursal que actualmente no tenga ningún empleado en la relación ***EmpleadoSucursal*** es necesario introducir valores nulos en los atributos relativos al empleado, sin embargo, como NumEmp es la clave primaria de la relación ***EmpleadoSucursal***, si tratamos de introducir un valor nulo en NumEmp se viola la integridad referencial.

Anomalía de eliminación

La anomalía de eliminación es lo opuesto de la anomalía de inserción. Se refiere a un caso en el que la eliminación de un dato de una entidad especifica provoca una perdida no intencionada de los datos que caracterizan a otra entidad.

Por ejemplo, si borramos una tupla de la relación ***EmpleadoSucursal*** que represente el último empleado de una cierto sucursal, también perderemos la información acerca de dicha sucursal. Como el caso de empleado SA9 (Sofia Sambrana) de la relación ***EmpleadoSucursal***.

Anomalía actualización

La anomalía de actualización surge cuando en la actualización de un valor de datos este requiere que se actualizar en muchas tuplas de la relación. Por ejemplo, si queremos cambiar el valor de la dirección de la sucursal B003, tenemos que actualizar las tuplas de todos los empleados de dicha sucursal.

3.2. Dependencias funcionales

Unidad IV: Lenguaje de Base de Datos

VII. Lista de referencias

Básicos

Abraham Silberschatz , Henry F. Korth y S. Sudarshan (2007). Fundamentos de Bases de datos. McGraw Hill, 5ta Edición, España

Thomas M. Connolly, Carolyn E. Begg(2005) Sistemas de bases de datos. Pearson-Hal, 4to Editorial, España

Date C.J. (2001) Introducción a los sistemas de base de dato. Editorial: Person Prentice Hall, México.

Complementarios

Kaoenke David M. (2003). Procesamiento de base de datos fundamento, diseño e implementación 8vo edición. México D.F: Pearson Education.

Rob, P., & Coronel, C. (2003). Sistemas de bases de datos: Diseño, implementación y administración, 5th ed.. Mexico City: Cengage Learning.
Retrieved from
<http://go.galegroup.com/ps/i.do?p=GVRL&sw=w&u=ueaccn&v=2.1&it=aboutBook&id=GALE|3BQN>

VIII. Glosario

A

Anomalías

B

Bases de dato:

C

Cardinalidad:

I

Inconsistencias:

R

Redundancia

Relación

Registro:

S

SQL: El lenguaje estructurado de consulta utilizado para realizar consultas en las bases de datos relacionales

T

Túplas:

IX. Anexo