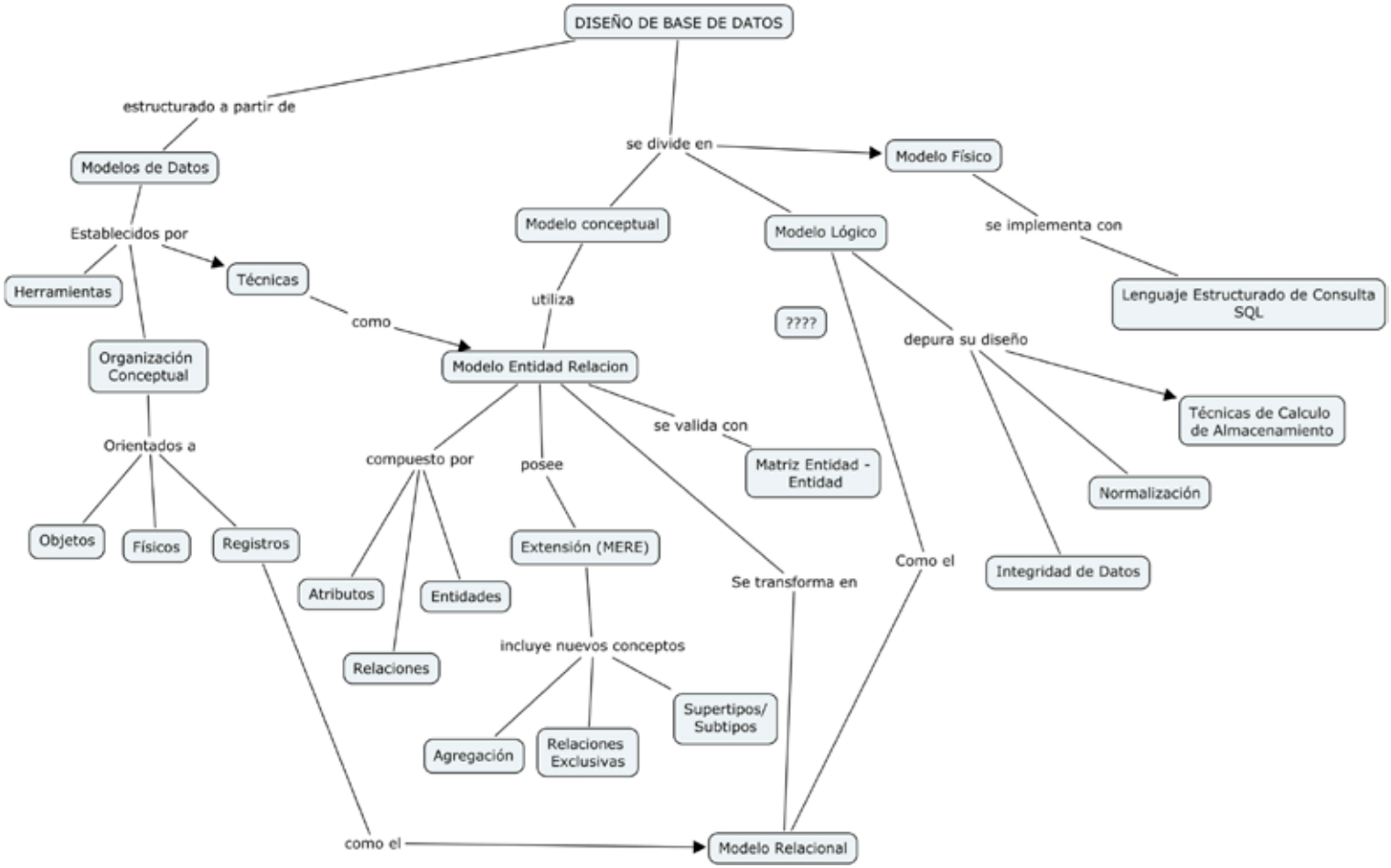
DISEÑO DE BASE DE DATOS

Introducción 3
SISTEMAS DE BASES DE DATOS 3
MODELAMIENTO DE DATOS 4
MODELOS DE DATOS 4
Modelos lógicos basados en objetos 4
Modelos lógicos basados en registros 5
Modelos físicos de datos 7
MODELO DE ENTIDAD RELACIÓN 8
Pasos para la construcción de un MER 9
Representación Gráfica del Modelo 11
Matriz Entidad - Entidad 13
MODELO ENTIDAD RELACIÓN EXTENDIDO 14
MODELO RELACIONAL 16
Normalización 18
Reglas de Integridad (Constraints) 18
Almacenamiento 21
BIBLIOGRAFÍA 24
GLOSARIO 25

Mapa conceptual
Diseño de Base de Datos



CONCEPTOS GENERALES DE DISEÑO DE BASE DE DATOS

INTRODUCCIÓN

Una de las tareas más importantes para un Administrador de Bases de Datos es la del diseño de base de datos, que abarca desde el modelamiento de datos en el ámbito lógicoconceptual, hasta la definición de las estructuras de almacenamiento, para lo cual se debe poseer total claridad y entero conocimiento del modelo de negocios, requerimientos de información y de los recursos e infraestructura con la que cuenta la organización tanto para el almacenamiento como para su procesamiento.

SISTEMAS DE BASE DE DATOS

Un sistema de base de datos está compuesto por cuatro elementos fundamentales: Hardware, Software, Usuarios y Datos, su objetivo principal está centrado en dar soporte al almacenamiento y procesamiento de datos y para esto se centra en la disminución de los siguientes factores:

- Redundancia e inconsistencia de datos.
- Dificultad para tener acceso a los datos.
- Aislamiento de los datos.
- Anomalías del acceso concurrente.
- Problemas de seguridad.
- Problemas de integridad.

Al diseñar una Base de Datos, su naturaleza y complejidad puede variar notoriamente. Independientemente del tamaño, complejidad de procesos involucrados y metodología de desarrollo, se tienen una serie de tareas estandarizadas:

- Recolección de Información
- Selección del modelo de datos.
- Definición de objetos, atributos y relaciones.
- Definición de Reglas de Integridada aplicar de acuerdo con los requerimientos del negocio y del modelo de datos adoptado.

MODELAMIENTO DE DATOS

MODELOS DE DATOS

El modelo de datos se define como una colección de herramientas conceptuales para describir los datos, sus relaciones, la semántica asociada a ellos y sus restricciones de consistencia.

Un modelo de datos, representa la organización conceptual o lógica de los datos que soportan las operaciones que se ejecuten en un sistema determinado.

Cada uno de los procesos de un sistema requiere de un conjunto de datos que soporten las transacciones a efectuar, de tal modo que un modelo de datos debe responder a: ¿Cómo se organizan los datos, de forma tal que su uso sea eficiente?.

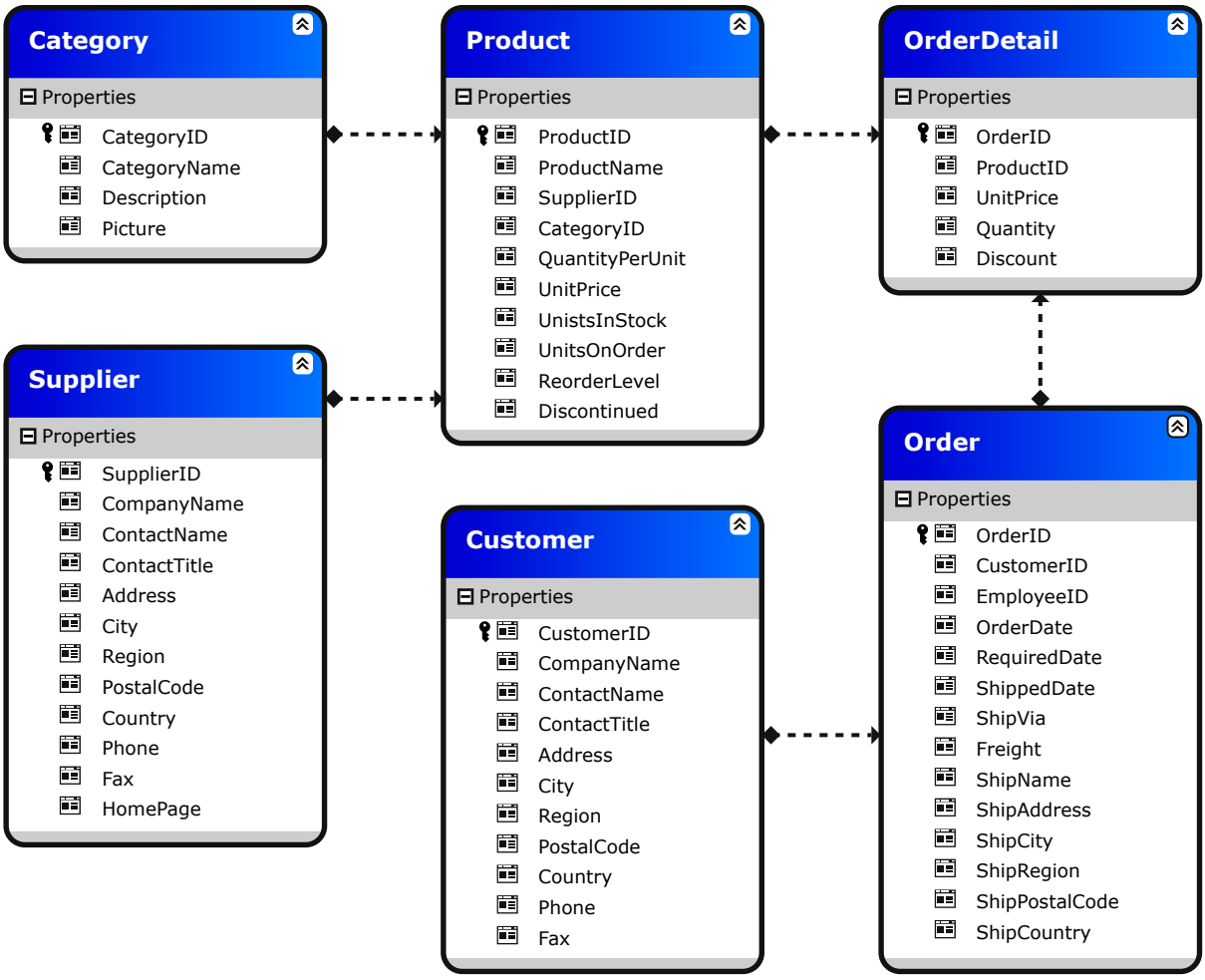
Un modelo establece una interpretación del mundo real al que se asocia, a través de una representación gráfica que relaciona los elementos que lo componen. Para responder a la acelerada transformación de la tecnología los modelos de datos han evolucionado y una clasificación que presenta los diferentes tipos de modelos de datos la constituyen los siguientes tres grupos:

Modelos lógicos basados en objetos, modelos lógicos basados en registros y modelos físicos de datos.

MODELOS LÓGICOS BASADOS EN OBJETOS

Estos modelos se usan para describir datos en el nivel conceptual y de visión. Su principal característica es la capacidad de estructuración ya que es bastante flexible y permiten que las restricciones de datos se realicen explícitamente. Algunos de los modelos más conocidos son;

- Modelo entidad-relación,
- Orientado a objetos
- Modelo funcional de datos.



MODELOS LÓGICOS BASADOS EN REGISTROS

Estos modelos se usan para describir datos en los modelos conceptual y físico. Permiten especificar la estructura lógica global de la Base de Datos y proporcionan una descripción a nivel más alto de la implementación.

Los tres modelos de datos más aceptados son los modelos: relacional, de red y jerárquico. Siendo el modelo relacional el más utilizado.

- Modelo relacional.

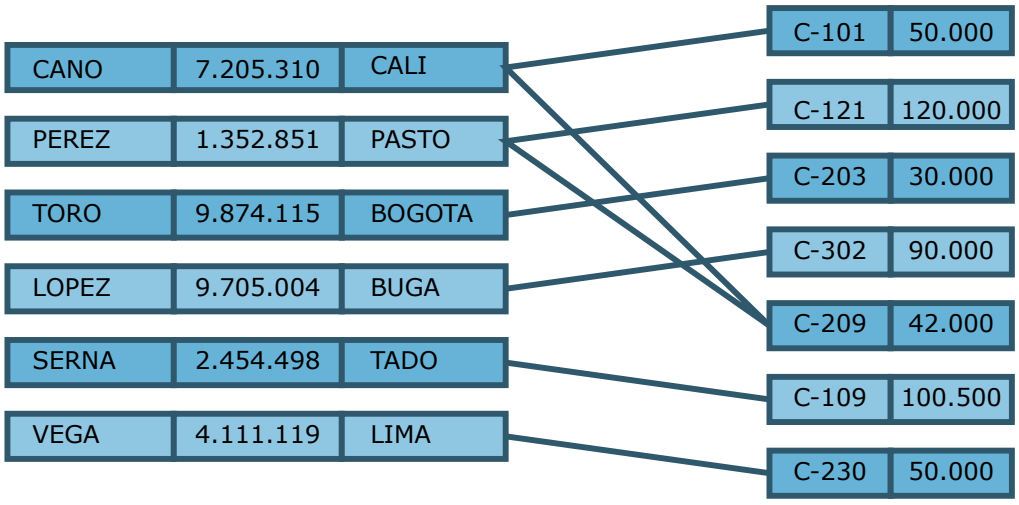
CLIENTE

NOMBRE	CEDULA	CUENTA	CIUDAD
CANO	7.205.310	C-101	CALI
PEREZ	1.352.851	C-121	PASTO
TORO	9.874.115	C-203	BOGOTA
LOPEZ	9.705.004	C-302	BUGA
SERNA	2.454.498	C-109	TADO
VEGA	4.111.119	C-230	LIMA
CANO	7.205.310	C-309	CALI
PEREZ	1.352.851	C-209	PASTO

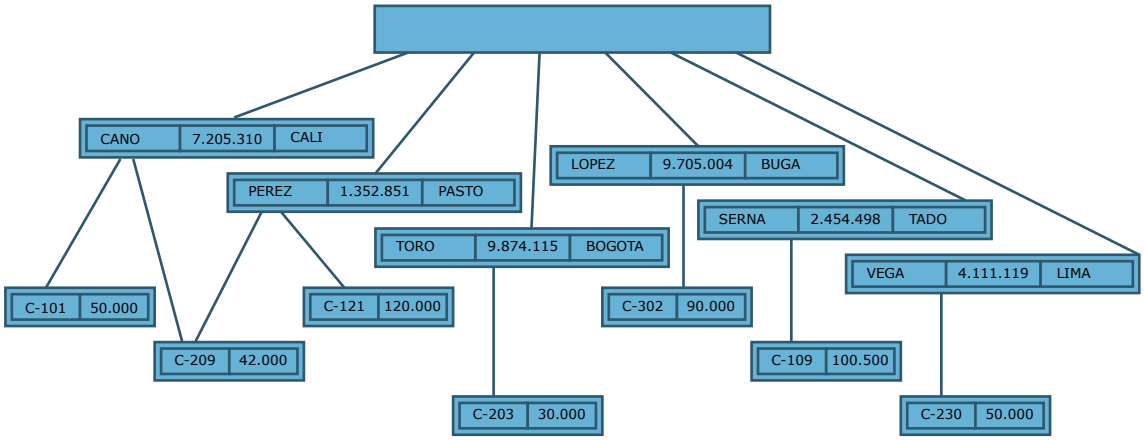
CUENTA

CUENTA	CUENTA
C-101	50.000
C-121	120.000
C-203	30.000
C-302	90.000
C-209	42.000
C-109	100.500
C-230	50.000

- Modelo de Red.

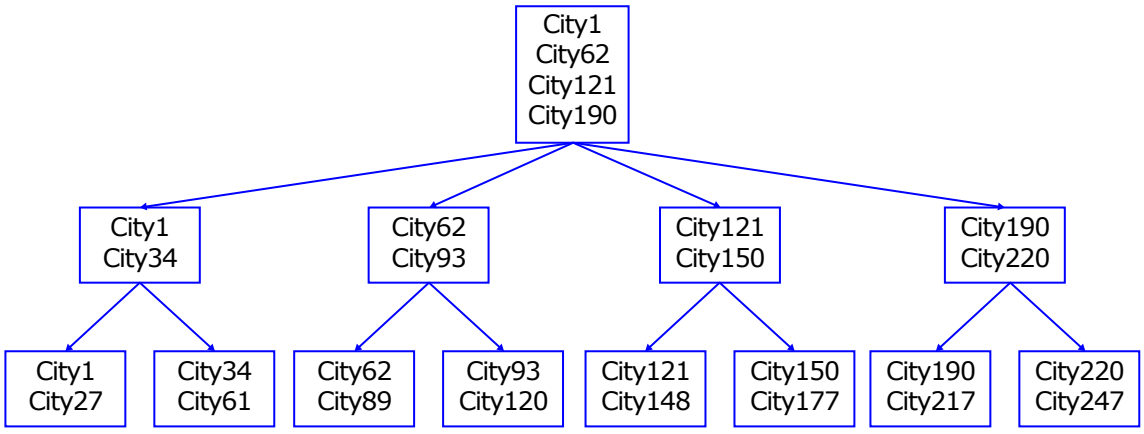


- Modelo Jerárquico.



MODELOS FÍSICOS DE DATOS

Estos modelos se usan para describir datos en el nivel más bajo. Hay muy pocos modelos físicos de datos en uso, siendo los más conocidos el modelo unificador y el de memoria de elementos. En estos casos las estructuras se implementan dentro del propio manejador de datos, por ejemplo: los Árboles binarios y las tablas Hash.



MODELO ENTIDAD RELACIÓN



El Modelo Entidad Relación (MER) es una metodología de diseño de Bases de Datos que consiste en representar a nivel conceptual los datos que soportan el funcionamiento de un sistema. El MER fue introducido originalmente por Peter Chen en 1976 y aunque ha sufrido variaciones en cuanto a los diagramas utilizados para representar sus elementos, su operación y utilidad siguen vigentes.

La base del MER está en identificar los elementos o entes importantes del sistema, los datos que componen cada uno de ellos y la interacción entre dichos elementos.

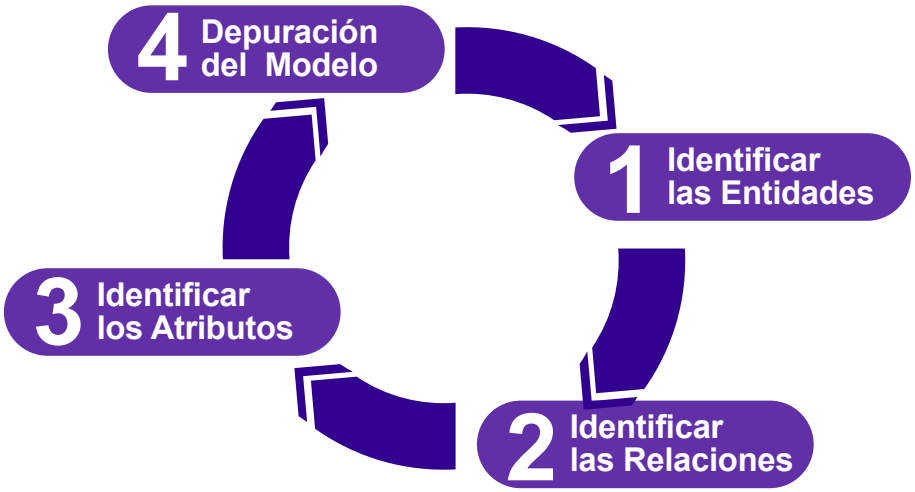
Los componentes básicos de un MER son: Entidades, Atributos y Relaciones.

ENTIDAD: Se denomina entidad a todo ente (conceptual o físico) del cual se desea mantener información.

ATRIBUTO: Conjunto de elementos de información que caracterizan a una entidad, identificándola, calificándola, cuantificándola, o declarando su estado.

RELACIONES: Identifica la interacción que existe entre dos entidades.

PASOS PARA LA CONSTRUCCIÓN DE UN MER

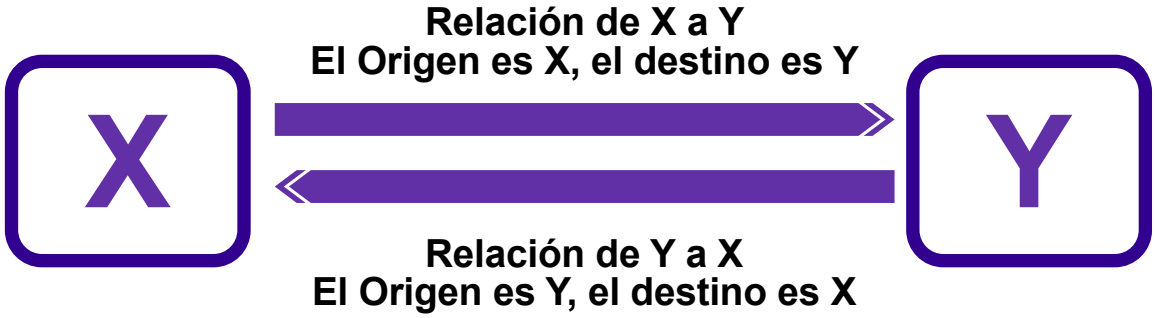


1. Identificar las Entidades

Inicialmente se requiere detectar los elementos u objetos de los cuales se requiere mantener información, a continuación identificar las operaciones que se ejecutan sobre estos elementos con lo cual es posible que aparezcan más entidades, pero solo deben quedar las que estén directamente relacionadas con el proceso a modelar. Los nombres de las entidades se deben escribir en mayúscula y singular.

2. Identificar las Relaciones

La interacción entre las entidades se define teniendo en cuenta que debe existir un análisis encada sentido de la relación para definir adecuadamente la cardinalidad (uno o varios) y el grado de participación (obligatoria u opcional) de las entidades en la relación.



Se debe evitar relacionar todas las entidades entre sí, por eso un error común en este proceso es tratar de relacionar todas las entidades entre sí, el consejo para evitar este error es pensar primero en las relaciones más evidentes y si se generan dos caminos distintos evaluar si alguno de ellos es redundante y desecharlo.

3. Identificar los atributos

Describir para cada entidad los atributos que la identifican (clave o llave) y constituyen, es importante tener claramente definido el tipo de dato al que pertenece, es importante definir si su valor es obligatorio u opcional, debe evitarse asignar un doble significado, solo se debe permitir el almacenamiento de un único dato, también verificar que no exista parcial o totalmente en otra entidad. Para cada una de estas características de los atributos existe una representación gráfica de acuerdo con la notación utilizada para realizar el diagrama.

Es posible que algunas relaciones posean atributos propios, en cuyo caso deben identificarse en el modelo.

Para establecer la clave se deben tener en cuenta los siguientes aspectos:

- Verificar si existen atributos en la entidad que identifiquen cada ocurrencia en forma única, es decir que sean candidatos para formar la clave. En caso de no existir se debe agregar un atributo para ser la clave, generalmente son números consecutivos que el SMBD puede controlar.

- Verificar que los atributos que forme la clave no sean opcionales, ya que un valor nulo, no es un dato válido para identificar una ocurrencia.

- Cuando la clave está compuesta por más de un atributo no se debe exceder en el número de ellos, de otra manera es más conveniente agregar un nuevo atributo.

4. Depuración del modelo

Una vez se ha definido un modelo inicial, se revisan las características de operación de las entidades desde el punto de vista de los datos asociados a la misma. Estos datos en algún momento del proceso toman valores definidos y es loque se conoce con el nombre de ocurrencia.

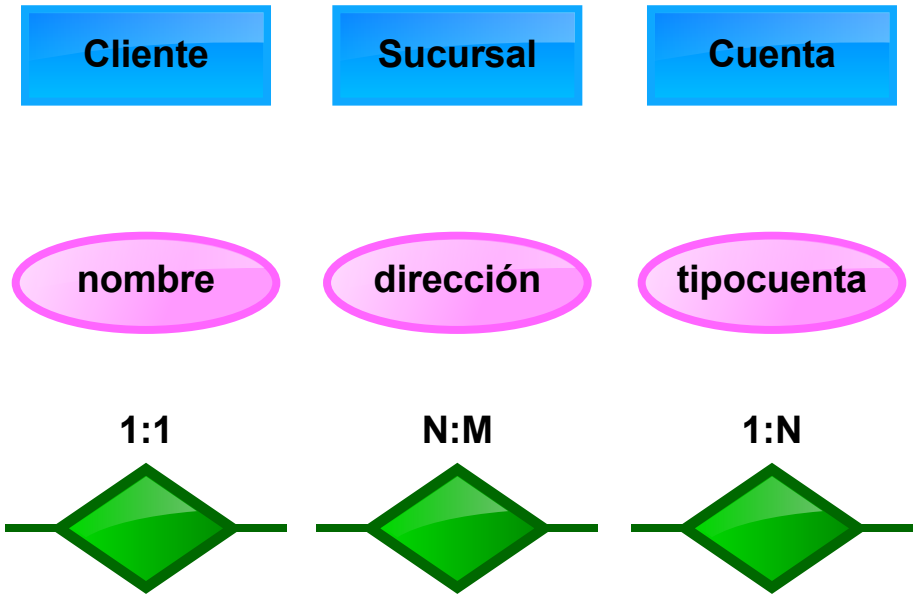
Por tanto se debe verificar:

- **Unicidad de Ocurrencias:** No deben existir dos filas, registros o tuplas con la misma información.
- **Múltiples Atributos:** Se deben tener al menos dos o más atributos en cada entidad.
- **Múltiples Ocurrencias:** Se deben evitar entidades que solo posean una ocurrencia (fila, tupla o registro)
- **Exclusividad de ocurrencias y atributos:** Una ocurrencia no puede estar contenida (total o parcialmente) en otra entidad. Se debe analizar el caso específico bien sea para fusionar las entidades, crear nuevas entidades o restablecer atributos.

REPRESENTACIÓN GRÁFICA DEL MODELO

Se han construido diferentes notaciones a través de los años, pero la esencia del modelo es la misma, a continuación se presentan algunas de estas para su comparación y análisis:

Notación de Peter Chen:



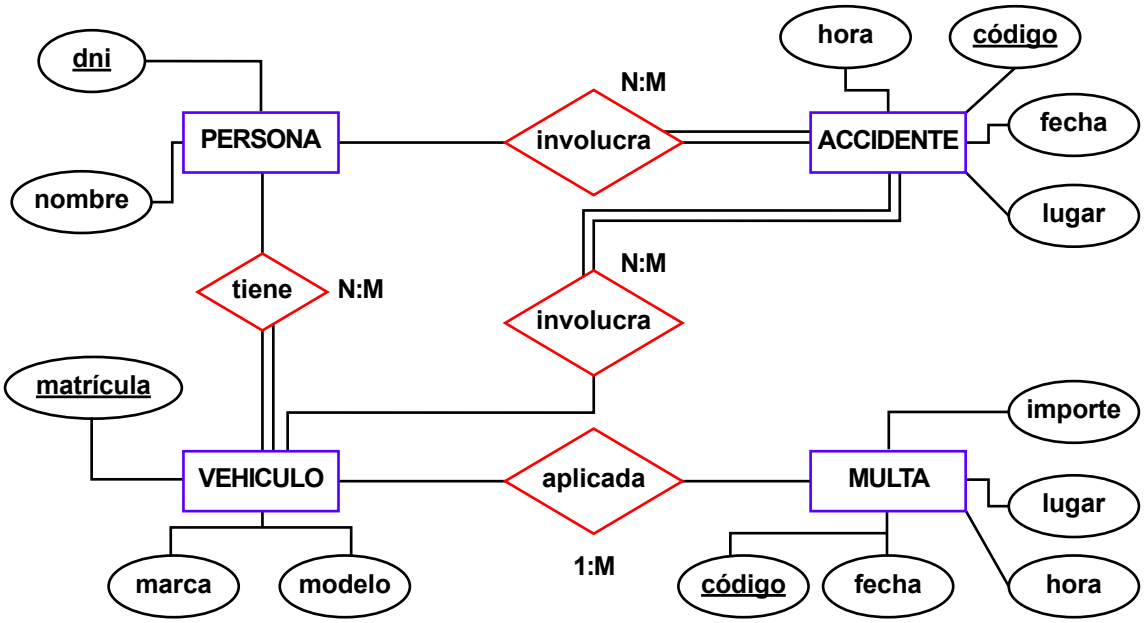


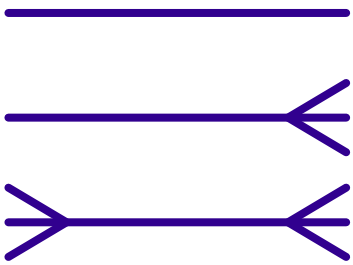
Figura 2. Modelo Entidad Relación con Notación de Peter Chen

NOTACIÓN CASE

Entidades y Atributos:



Relaciones:



(1 a 1)

(1 a muchos)

(muchos a muchos)

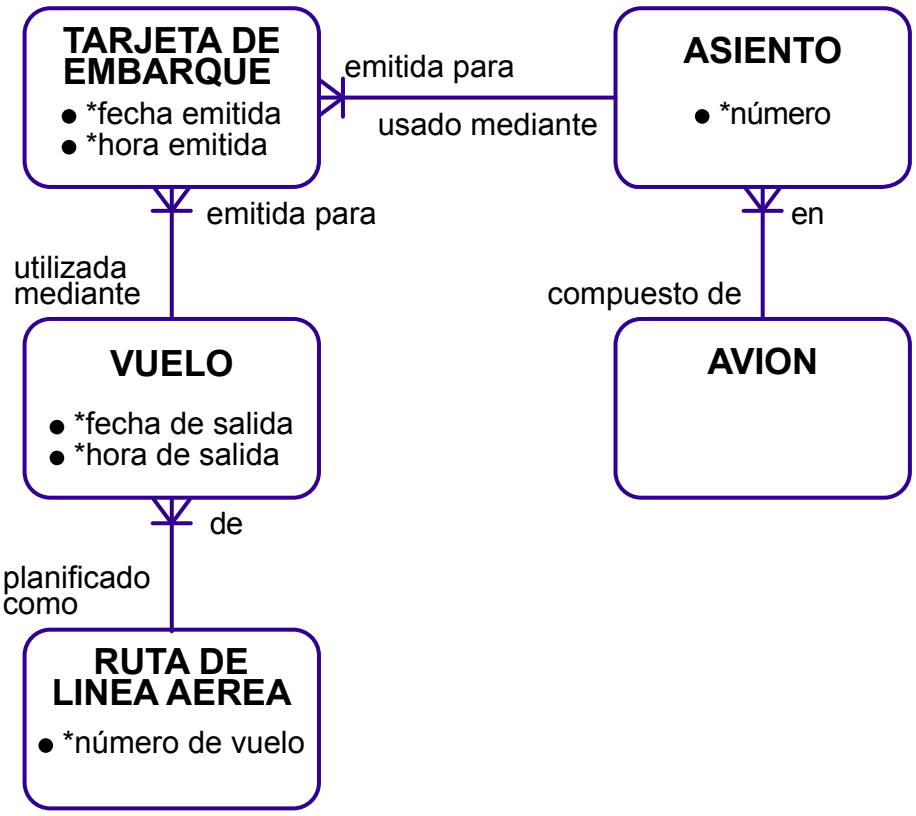


Figura 3. Modelo Entidad Relación con Notación Case Method

MATRIZ ENTIDAD – ENTIDAD

Para validar el MER es posible utilizar una matriz Entidad x Entidad, donde se observan las entidades y sus relaciones. Para construirla se hace una matriz con las entidades en filas y columnas, en cada intersección se escribe el nombre de la relación en singular, indicando además la cardinalidad y la obligatoriedad usando pares ordenados (el primer término indica opcional -0- u obligatorio -1- y el segundo indica cardinalidad uno -1- ó varios -n-). En la casilla donde se crucen dos entidades sin relación alguna se coloca una "X".
La siguiente representa la matriz entidad-entidad de un modelo propuesto para controlar la realización de eventos en una institución.

	EVENTO	TIPO	PARTICIPANTE	INSTITUCION
EVENTO	X	Pertenece (1,1)	Se inscriben (1,n)	Los realiza (1,n)
TIPO	Pertenecen (0,n)	X	X	X
PARTICIPANTES	Se inscribe (1,n)	X	X	X
INSTITUCION	Realiza (0,n)	X	X	X

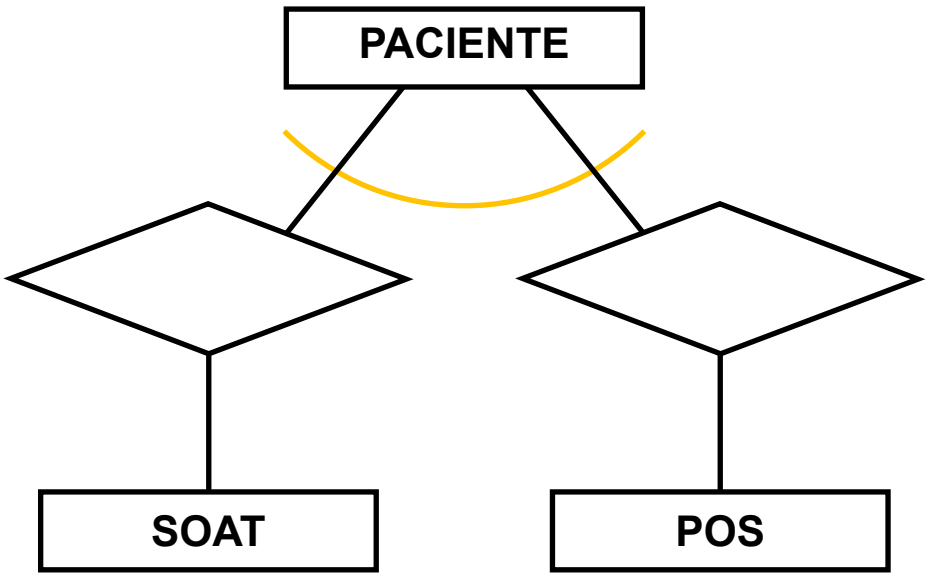
Tabla 1. Ejemplo de Matriz Entidad-Entidad

MODELO ENTIDAD RELACIÓN EXTENDIDO

Debido a la dificultad del modelo básico para representar ciertas situaciones, se crea el modelo entidad relación extendido -MERE- que recoge 3 elementos nuevos:

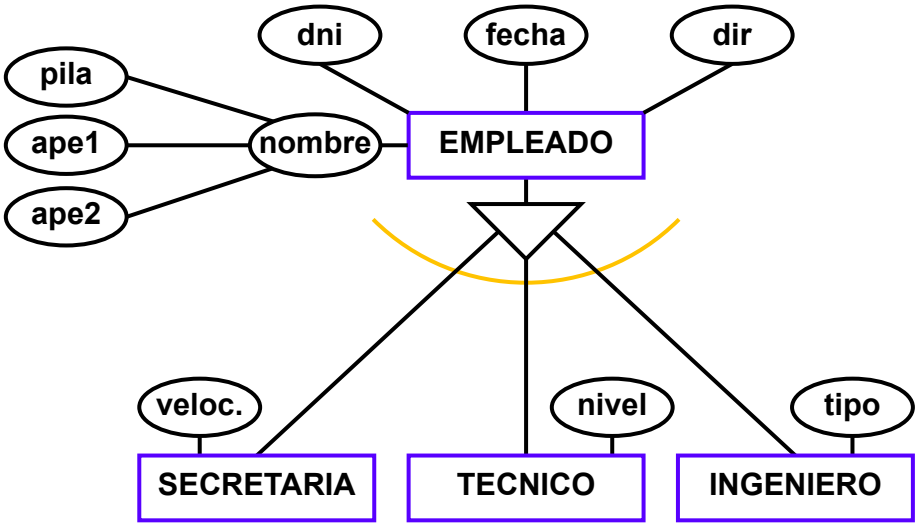
- **Relaciones excluyentes:**

  Cuando una entidad mantiene relación con otras pero esto no se puede hacer simultáneamente, ya que en cada instancia hay relación con una sola.



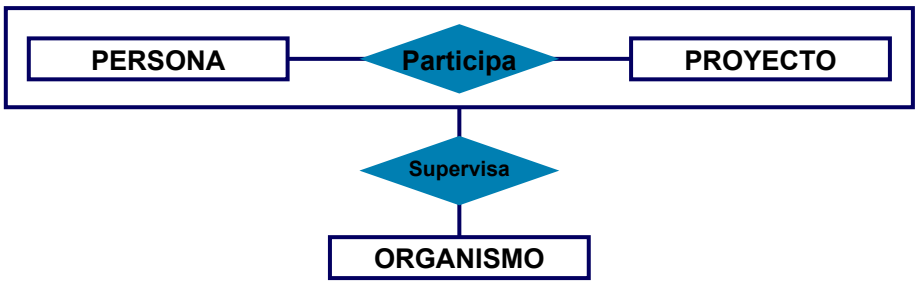
- **Especificaciones de Generalización y Especialización (supertipos y subtipos):**

  Cuando se identifican grupos de instancias con características básicas o atributos en común.

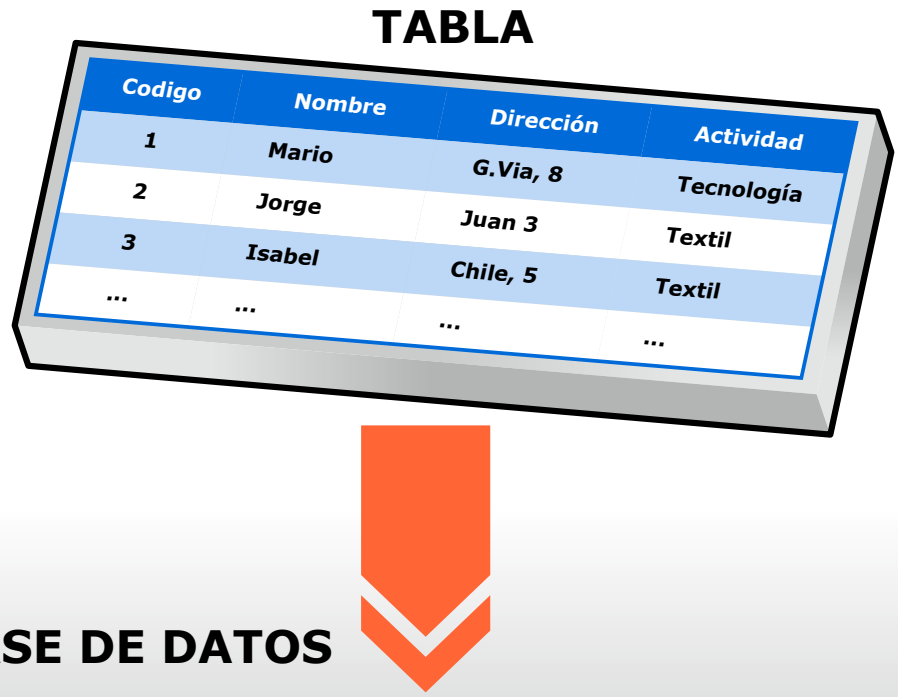


- **Agregación:**

  Para representar relaciones de grado 3 o cuando una relación a su vez se relaciona con otra entidad.



MODELO RELACIONAL



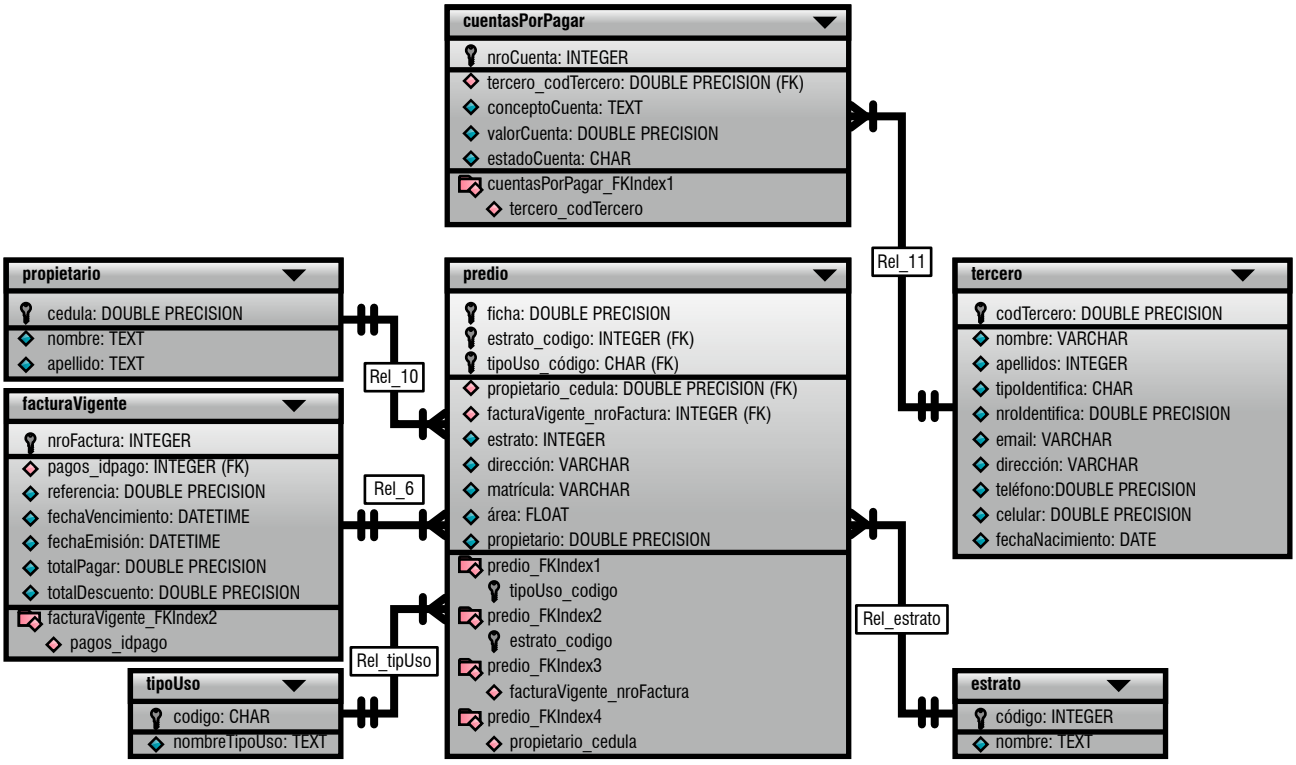


Figura 4. Modelo Relacional

Es un modelo formal propuesto por E.W. Codd en 1970, basado en los conceptos matemáticos de Relación y Teoría de Conjuntos. Este modelo esta soportado por la mayoría de los productos de bases de datos en el mercado actual y se representa por tablas en forma bidimensional.

Características:

- Los renglones, líneas o filas, poseen datos acerca de una entidad.
- Las columnas contienen datos acerca de los atributos de la entidad.
- Cada posición en la tabla almacena un valor simple, atómico, de un atributo.
- Todos los valores almacenados en una columna son del mismo tipo, es decir, están definidos sobre el mismo dominio.
- Cada columna o atributo debe poseer un nombre único.
- El orden de las columnas no es relevante.
- El orden de las filas no es relevante.
- No pueden existir dos filas idénticas en la tabla.

Es posible transformar el Modelo Entidad Relación (MER) al Modelo Relacional, a partir de la utilización de una serie de reglas, algunas de estas son:

- Toda entidad se transforma en una tabla.
- Todo atributo se transforma columna o campo dentro de una tabla.
- El identificador único de la entidad se convierte en clave primaria de la tabla.
- Aplicar la preparación de los esquemas entidad-relación mediante la aplicación de las reglas que faciliten y garanticen la fiabilidad del proceso de transformación, las cuales se basan en la Normalización.
- Realizar la Transformación de relaciones, donde es necesario verificar la cardinalidad y la verificación de si la relación es obligatoria u opcional.

NORMALIZACIÓN

La normalización es un proceso que permite garantizar un mínimo de redundancia y dependencias incoherentes sobre los datos, consiste en organizar los datos en tablas y establecer relaciones entre las tablas según reglas predefinidas.

La redundancia sobre los datos crea problemas de mantenimiento en la base de datos y ocasiona desperdicio de espacio en disco. Si hay que cambiar datos que existen en más de una tabla, se hace necesario hacer la actualización en todas sus ubicaciones. Además se generan problemas en las inserciones de nuevos datos, consultas y eliminaciones.

El proceso consiste en aplicar reglas de normalización sobre las tablas de una base de datos, cada regla se denomina "Forma Normal". Si una tabla cumple la primera regla, se dice que está en la "primera forma normal" y si cumple la regla N esta en forma normal N. Aunque son posibles otros niveles de normalización, la tercera forma normal se considera el máximo nivel necesario para la mayor parte de las aplicaciones.

REGLAS DE INTEGRIDAD (CONSTRAINTS)

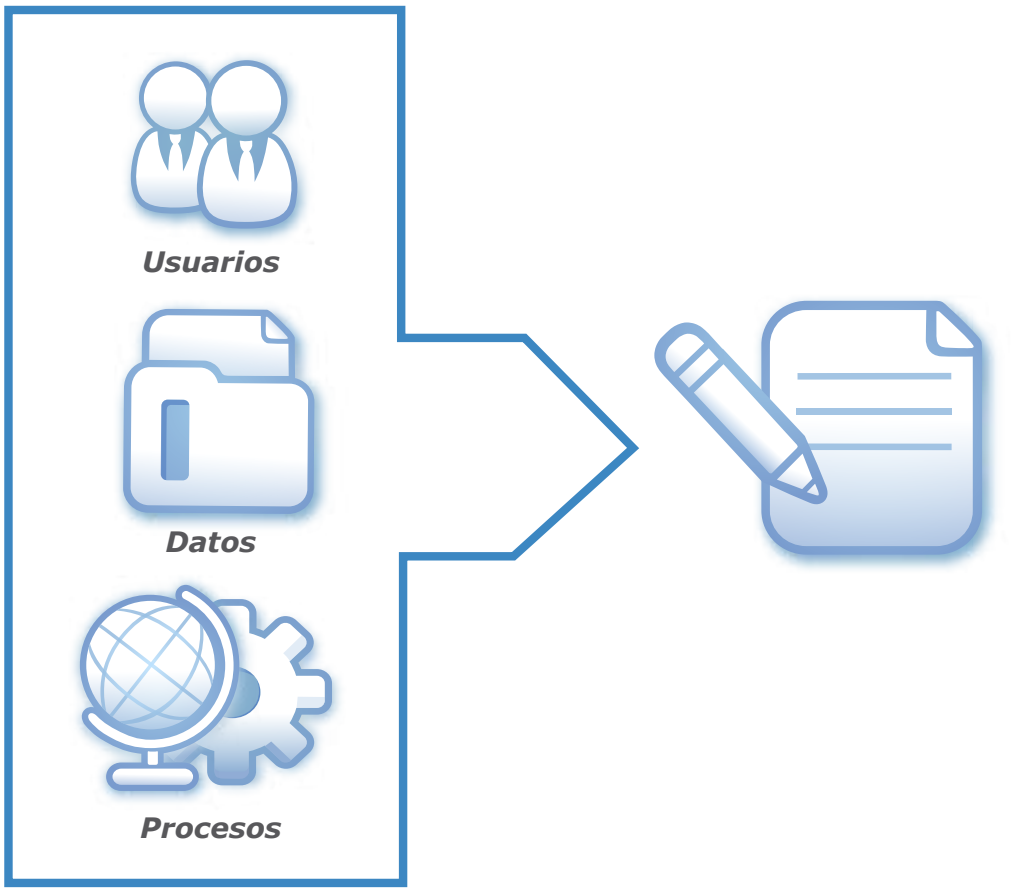
Aun cuando no se puede garantizar que los datos sean fidedignos por cuanto dependen en su gran mayoría de los usuarios, debemos apoyar la funcionalidad y cumplimiento de algunas reglas que puedan garantizar su Integridad.

En general, las condiciones que garantizan la integridad de los datos pueden ser de dos tipos:

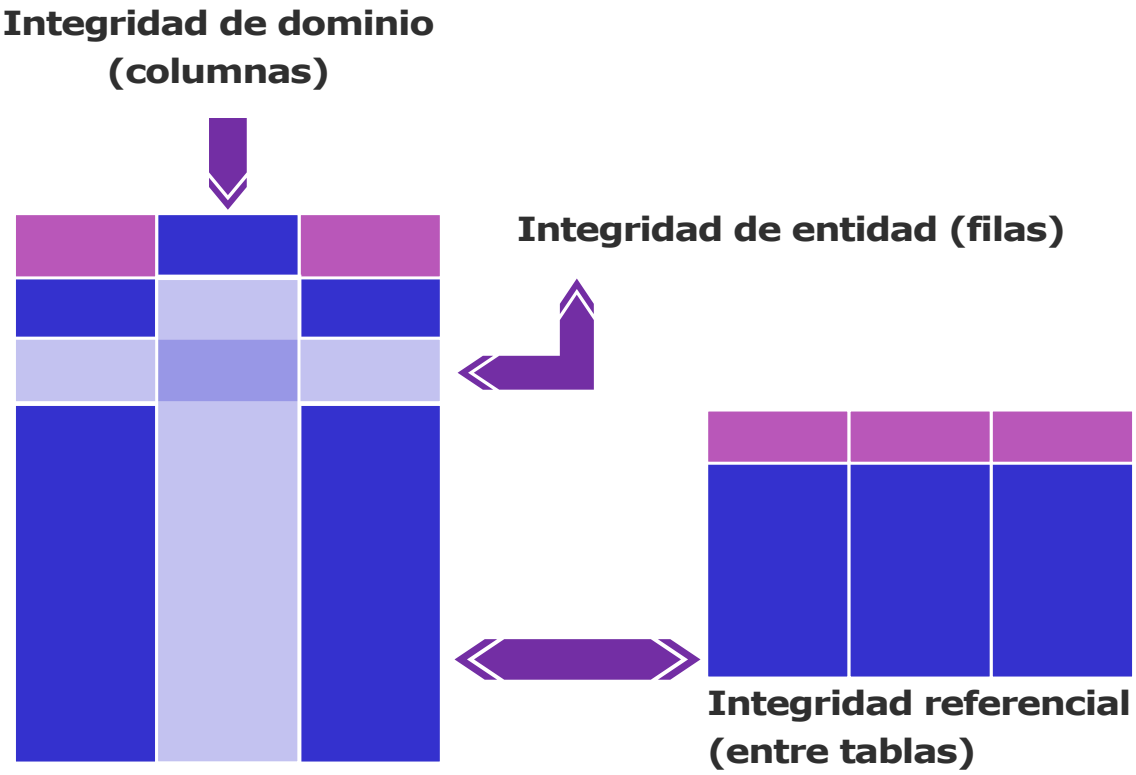
1. Las restricciones de integridad de usuario:

Están asociados a las reglas del negocio, son condiciones específicas de una base de datos concreta que no son necesariamente relevantes en otra base de datos.

Están determinadas por la funcionalidad de los procesos sobre los que se va a manipular y por las validaciones mínimas que se esperan sobre los datos.



2. Las reglas de integridad de modelo. Son las condiciones generales de un modelo de datos. Estás son genéricas y en el caso del modelo relacional son:



- **Unicidad de la clave Primaria:** toda clave primaria que se defina no debe admitir valores repetidos.
- **Integridad de entidad de la clave primaria:** los atributos de la clave primaria no pueden tener valores nulos.
- **Integridad Referencial:** Definición de las políticas de inserción y eliminación para las filas cuyos que tengan campos referenciados.
- **Integridad de Dominio:** Los valores almacenados en un atributo deben ser del dominio declarado para dicho atributo (tipo de datos, rango de valores admitidos, opcional/obligatorio, cumplimiento de condiciones especiales).

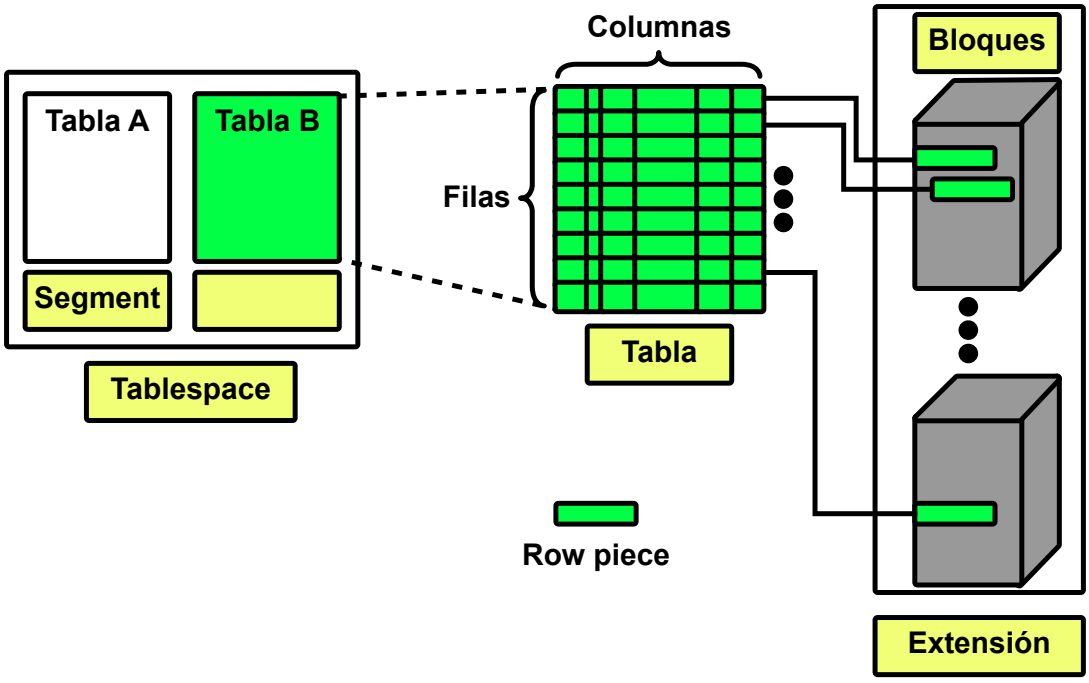
ALMACENAMIENTO

Uno de los problemas comunes a los que se enfrenta el Administrador de la Base de datos es el relacionado al tema del espacio que debe asignar a las bases de datos en los discos duros para no tener dificultades en su almacenamiento, funcionamiento y rendimiento.

Complementario al diseño dela base de datos, es necesario realizar una estimación del tamaño que tendrá la base de datos cuando se encuentre en producción. Esta estimación puede ayudar a determinar la configuración de hardware que será necesaria para conseguir el rendimiento que se requiere asegurando el espacio en disco necesario para el almacenamiento de los datos y demás objetos asociados a las bases de datos.

Está estimación también ayuda a determinar si el diseño requiere ajustes, ya que si el espacio requerido es muy grande, se podría aplicar mayor nivel de normalización, o en caso contrario si el tamaño calculado es inferior al esperado, se puede reducir la normalización para la optimización de tiempos de ejecución de las consultas.

Para comenzar se puede realizar una estimación del tamaño de cada tabla por separado y sumar los valores obtenidos. Se debe tener en cuenta si se tiene índices y el tipo de índices también como parte del tamaño de la tabla.



Un cálculo base del espacio requerido lo entrega el modelado de las bases de datos al determinarse cuántos registros se esperan por cada tabla en un período determinado. Por ejemplo en un mes. Este valor se debe proyectar o multiplicar por el período de vida útil activa de la información por ejemplo 12 meses, 24, 36, etc.

El ejercicio se realiza para cada tabla u objeto de la base de datos. Al final se suman todos los datos y se obtiene una aproximación al tamaño de las bases de datos.

Por cada tabla se debe tener en cuenta el tamaño de cada atributo o columna, el cual depende a su vez del tipo de información que contenga. Está información se encuentra registrada en el diccionario de datos. Se debe verificar de acuerdo al Sistema Manejador de Base de Datos, cuántos bytes utiliza por cada tipo de dato. Por ejemplo, si el SMBD utiliza para un tipo Numeric 4 bytes y en la tabla hay tres campos Numeric, ya se tienen 12 bytes a ocupar.

Cuando el campo ha sido definido de tipo variable, se debe calcular, inicialmente, al máximo posible. Esto puede variar de acuerdo a la oportunidad del dato pero es mejor pecar por exceso que por defecto.

Nombre del Archivo: CatAdsc.dbf **Fecha de Creación:20/05/2008**
Descripción: Archivo para el catálogo de adscripciónes del instituto

Campo	Tipo	Tamaño	Descripción
cveDepto	Numérico	5	Clave de Adscripción
cDescripción	Caracter	75	Nombre de Adscripción

Relaciones: cveAdsc con BD Nomina
Campos Clave: cveAdsc.

Nombre del Archivo: CatDpto.dbf **Fecha de Creación:20/05/2008**
Descripción: Archivo para el catálogo de departamentos del instituto

Campo	Tipo	Tamaño	Descripción
cveDepto	Numérico	5	Clave de Departamento
cDescripción	Caracter	75	Nombre del Departamento

Relaciones: cveDpto con BD Nomina
Campos Clave: cveAdsc.

Nombre del Archivo: CatCat.dbf **Fecha de Creación:20/05/2008**
Descripción: Archivo para el catálogo de departamentos del instituto

Campo	Tipo	Tamaño	Descripción
cveDepto	Numérico	5	Clave de Catálogo de Burócratas
cDescripción	Caracter	75	Descripción del puesto de burócratas

Relaciones: cveCat con BD Nomina
Campos Clave: cveCat.

Diccionario de datos.

Es muy importante hacer la consulta de los tamaños que utiliza el Gestor de Base de Datos así como la forma en que almacena la información (por ejemplo campos Null, seguimiento interno de datos y cambios a los mismos, etc). La arquitectura de almacenamiento de cada SMBD influye también en la estimación de tamaños.

BIBLIOGRAFÍA

BARKER, R. (1990). Case*Method: Entity Relationship Modelling. Addison Wesley.

CASTAÑO, A. (2005). Diseño de Base de Datos Relacionales Distribuidas. Problemas Resueltos (pp. 429-458). México: Alfaomega Grupo Editor.

Microsoft. MSDN Library (en español). Estimar el tamaño de una base de datos. Recuperado en Marzo 20 de 2012, desde: http://msdn.microsoft.com/es-es/library/ms187445(v=sql.105).aspx

RAMIREZ CANO, M. E. (2005). Base de datos I. Centro de Publicaciones FUNDAUTONOMA.

Wiki de PostgreSQL. (2011, Enero 16). Recuperado en Abril 12 de 2012, desde: http://wiki.postgresql.org/wiki/Español

WikiLearning. (2009). Iniciación a Oracle - conceptos de almacenamiento. Recuperado en Marzo 15 de 2012, desde: http://www.wikilearning.com/curso_gratis/iniciacion_a_oracle/3861-6

Wikipedia. (2012). Normalización de bases de datos. Recuperado en Junio 12 de 2012, desde: http://es.wikipedia.org/wiki/Normalizaci%C3%B3n_de_bases_de_datos#Formas_Normales

GLOSARIO

Cardinalidad: Representa el numero de instancias de una entidad que puede estar presente en otra entidad relacionada.

Ocurrencia: Una instancia de datos, es decir una fila de datos en una tabla.

SQL: Lenguaje estructurado de Consulta que contiene una estructura estándar para la realización de operaciones sobre las bases de datos.

SGBD: Sistema de Gestión de Bases de Datos

Tupla: Colección única de datos resultante de una consulta en bases de datos. Se suele asociar a un registro o fila de una tabla de datos.

OBJETO DE APRENDIZAJE	Diseño de Base de Datos
Desarrollador de contenido Experto temático	**Julio César Hernández** **Ana Yaqueline Chavarro Parra**
Asesor Pedagógico	**Rafael Neftalí Lizcano Reyes**
Productor Multimedia	**José Jaime Luis Tang Pinzón** **Victor Hugo Tabares Carreño**
Programadores	**Daniel Eduardo Martínez Díaz** **Grateful Dead Montaño Sierra**
Líder expertos temáticos	**Ana Yaqueline Chavarro Parra**
Líder línea de producción	**Santiago Lozada Garcés**

Atribución, no comercial, compartir igual

Este material puede ser distribuido, copiado y exhibido por terceros si se muestra en los créditos. No se puede obtener ningún beneficio comercial y las obras derivadas tienen que estar bajo los mismos términos de licencia que el trabajo original.

SENA

Creative Commons