

• PRINCIPIOS DE LOS SISTEMAS DE BASES DE DATOS

• Base de Datos, Concepto.–

- Es una colección de datos interrelacionados almacenados en conjunto sin redundancias, cuya finalidad es la de servir a una aplicación o más, los datos se almacenan de modo que resulten independientes de los programas que los manipulan. En otras palabras el conjunto de datos se conoce con el nombre de base de datos.
- Es un conjunto autodescriptivo de registros integrados:

Autodescriptiva: además de los datos fuente del usuario contiene también una descripción de su propia estructura (diccionario de datos).

Registros Integrados: La jerarquía normal de los datos es la siguiente: los bits conforman bytes o caracteres; los caracteres constituyen campos; los campos integran registros y los registros componen archivos. Una BDD incluye **archivos** de datos del usuario y más, una descripción de los datos **metadatos**, índices para formar las relaciones entre los datos y la última categoría de datos se denomina **metadatos de aplicación** información de las aplicaciones que la utilizan.

Jerarquía de los elementos de datos: (a) En el procesamiento de archivos y (b) En el procesamiento de Bases de Datos.

• OBJETIVOS DE LA BASE DE DATOS

a.– Versatilidad para la representación de relaciones.– El sistema de administración de datos debe ser capaz de representar relaciones de los datos almacenados, y crear los archivos lógicos que se requieren.

b.– Desempeño

c.– Costo mínimo

d.– Redundancia Mínima

e.– Capacidad de Búsqueda.– La capacidad para explorar una base de datos rápidamente y con diferentes criterios de búsqueda depende mucho de la organización física de los datos

f.– Integridad.– Toda la instalación debe garantizar la integridad de la información almacenada

g.– Reserva Privacidad y Seguridad.– La reserva se refiere al derecho de los individuos y organismos para determinar por si mismos, Cuándo, Como y en qué medida se permitirá la transmisión a terceros de la

información que les concierne

h.- La interfase con el Pasado.- Cuando un organismo instala un nuevo software de la Base de Datos, es importante que este pueda trabajar con los programas, procedimientos existentes y que los datos ya almacenados puedan ser convertidos a las nuevas formas

i.- La interfase con el Futuro.- Es importante en el diseño de una Base de datos planearla de manera que sea modificable sin necesidad de cambiar los programas de aplicación en uso

j.- Afinación.- Es necesario ajustes y cambiar fundamentalmente la organización del almacén de datos después de que el sistema ha entrado en servicio y se han aclarado suficientemente las pautas de uso. Este proceso de ajuste de la base de datos se llama afinación (Tunning)

k.- Migración de Datos.- Es conveniente mudar un conjunto de datos dentro del almacén de datos a posiciones accesibles de acuerdo con su actividad. Este proceso de ajuste del almacenamiento de los datos se llama migración de Datos. En algunos sistemas esto se hace automáticamente en otros lo hacen los programadores del sistema o el administrador de datos.

El administrador de la base de datos se encarga de supervisar y mantener la vista lógica global de los datos.

l.- Simplicidad.- Los medios que se utilizan para representar la vista general de los datos deben ser concebidos de manera simple y nítido.

• TERMINOLOGIA RELACIONADA A UNA BASE DE DATOS.

a.- Item, Campo o Atributo.- es la cualidad o las características de un objeto y es la unidad de datos más pequeña.

b.- Registro.- (segmento o tupla), es una colección de items. (sinónimo de fila).

c.- Celda.- es el término genérico con que se refiere a una pista, un cilindro, un módulo o cualquier área de almacenamiento, es el valor de un atributo en determinada ocurrencia.

d.- Ocurrencia.- es el conjunto de registros que cumplen determinado criterio.

e.- Set.- es la relación definida entre dos miembros o entidades.

f.- Cursor.- es el apuntador a un conjunto de registros o ocurrencias con determinado criterio en la base de datos.

g.- Privilegios.- son las cualidades que se le asignan a cada uno de los atributos de una base de datos para su acceso a él.

h.- Tabla.- es un registro diferente o un tipo de registro que difiere de otro. También se puede decir que es una estructura que permite almacenar una entidad.

i.- Query.- Son consultas que se realizan a la base de datos.

j.- Diccionario de Datos.- es un lugar en donde se almacena la información relativa a la estructura de la base de datos. En otras palabras es la colección de toda la información para mantener una base de datos como: campos accesados, índices usados, usuarios activos, etc.).

• ABSTRACCIÓN DE DATOS

Un sistema de gestión de bases de datos es una colección de archivos interrelacionados y un conjunto de programas que permiten a los usuarios acceder y modificar esos archivos. Un objetivo importante de un SDBDD es proporcionar a los usuarios una *visión abstracta* de los datos. Es decir, el sistema esconde ciertos detalles de cómo se almacenan y mantienen los datos.

Existen tres niveles de abstracción:

- **Nivel físico.** El nivel más bajo de abstracción describe cómo se almacena realmente los datos. En el nivel físico, se describen en detalle las estructuras de datos complejas del nivel bajo.
- **Nivel conceptual.** El siguiente nivel más alto de abstracción describe qué datos son realmente almacenados en la base de datos y las relaciones que existen entre los datos. Aquí se describen la base de datos completa en términos de un número pequeño de estructuras relativamente. Aunque la implementación de las estructuras sencillas del nivel conceptual puede implicar estructuras complejas del nivel físico, el usuario no necesita darse cuenta de esto. Este nivel es usado por los administradores de bases de datos, quienes deben decidir qué información se va a guardar en la base de datos.
- **Nivel de visión.** El nivel más alto de abstracción describe sólo parte de la base de datos completa. A pesar del uso de estructuras más sencillas en el nivel conceptual, permanece algo de complejidad debido al gran tamaño de la base de datos. Muchos usuarios del sistema de bases de datos no se interesarán por toda la información. En cambio, dichos usuarios sólo necesitan una parte de la base de datos. Para simplificar su interacción con el sistema, se define el nivel de abstracción de visión. El sistema puede proporcionar muchas visiones de la misma base de datos.

2.5. MODELO DE DATOS

El modelo de datos es un grupo de herramientas conceptuales que permiten describir los datos, las relaciones, su semántica y sus limitaciones.

Se dividen en tres grupos:

- a) Modelos lógicos basados en objetos
- b) Modelos lógicos basados en registros
- c) Modelos físicos de datos

2.5.1. MODELOS LÓGICOS BASADOS EN OBJETOS

Se utilizan para describir los datos en los niveles conceptual y de visión. Se caracterizan por el hecho de que permiten una estructuración bastante flexible y hacen posible especificar claramente las limitantes de los datos. Dentro de este tipo los modelos más conocidos son:

- Modelo entidad–relación
- Modelo orientado a objetos
- Modelo binario
- Modelo semántico de datos
- Modelo infológico
- Modelo funcional de datos

El modelo E–R, es el más apropiado para el diseño de las bases de datos; es decir, es el que mejor se presta para la transformación del modelo conceptual al diseño lógico de la base de datos.

2.5.2. MODELOS LÓGICOS BASADOS EN REGISTROS

Se utilizan para describir los datos en los niveles conceptual y físico. A diferencia de los modelos de datos basados en objetos, estos modelos sirven para especificar tanto la estructura lógica general de la base de datos como una descripción a un nivel más alto de la implementación, sin embargo no permiten especificar en forma clara las limitantes de los datos.

Los tres modelos de datos más ampliamente aceptados son:

- Modelo Jerárquico
- Modelo Red o Plex
- Modelo Relacional

Modelo Relacional

En este modelo los datos y las relaciones entre los datos se presentan por medio de una serie de tablas, cada una de las cuales tiene varias columnas con nombres únicos y una columna de una tabla representa una relación entre un conjunto de valores.

Modelo Red

Este tipo de organización se la conoce como CODASYL(Conference On Data Systems Languages), esta

representado por medio de un conjunto de registros y las relaciones entre los datos se representan por medio de ligas, que pueden considerarse como apuntadores. Los registros de la base de datos se organizan en forma de conjuntos de gráficas arbitrarias, es decir un hijo puede tener más de un padre.

Ejm:

%%%%%%%%%% %%%%%%%%%%

% REVISTAS % % BEBIDAS %

%%%%%%%%%% %%%%%%%%%%

% %%% %

% %%% %

%%%%%%%%%% % %%%%%%%%%%

%CLIENTE 1 %%% % CLIENTE 2 %

%%%%%%%%%% %%%%%%%%%%

Modelo Jerárquico

El modelo Jerárquico es similar al modelo de red en cuanto a que los datos y las relaciones entre los datos se representan por medio de registros y ligas (una liga es una asociación entre dos registros), el modelo jerárquico difiere al modelo red en que los registros están organizados como conjuntos de árboles en lugar de gráficas arbitrarias.

• GESTOR DE BASES DE DATOS

Un gestor de base de datos es un módulo de programa que proporciona el interfaz entre los datos de bajo nivel almacenados en la BDD y los programas de aplicación y consultas hechos al sistema. El GBDD es responsable de las siguientes tareas.

- **Interacción con el gestor de archivos:** el GBDD traduce las distintas sentencias DML a comandos del sistema de archivos de bajo nivel. Así, el GBDD es el responsable del verdadero almacenamiento, recuperación y actualización de los datos en la BDD.
- **Implantación de la Integridad:** los valores de los datos que se almacenan en la BDD deben satisfacer ciertos tipos de restricciones de consistencia.
- **Implantación de la seguridad:** no todos los usuarios de la BDD necesitan tener acceso a todo su contenido, el GBDD debe hacer cumplir estos requisitos de seguridad.
- **Copia de seguridad y recuperación:** Podría existir fallas externas al sistema que provocarían pérdida de la información por ejemplo daño físico del disco, problemas de suministro de energía y errores de software. Es responsabilidad del GBDD detectar tales fallos y restaurar la información de la BDD al estado que tenía antes de ocurrir el fallo.

- **Control de recurrencia:** cuando varios usuarios actualizan la BDD concurrentemente, es posible que no se conserve la consistencia de los datos. Controlar la interacción entre los usuarios concurrentes es otra responsabilidad del gestor de la BDD.

• ADMINISTRADOR DE BASE DE DATOS

Una de las razones principales para tener sistemas de gestión de BDD es tener control central de los datos y de los programas que accedan a esos datos. La persona que tiene dicho control central sobre el sistema se llama administrador de la BDD (database administrator (DBA)). Las funciones del DBA incluyen:

- **Definición de esquema:** al compilar las sentencias DDL resultan tablas que son almacenadas permanentemente en el diccionario de datos.
- **Definición de la estructura de almacenamiento y del método de acceso:** Estructuras de almacenamiento y métodos de acceso adecuados se crean escribiendo un conjunto de definiciones que son traducidas por el compilador del lenguaje de almacenamiento y definición de datos.
- **Modificación del esquema y de la organización física:** las modificaciones, tanto al esquema de la BDD como a la descripción de la organización física de almacenamiento, se logran escribiendo un conjunto de definiciones que son usadas bien por el compilador de DDL o bien por el compilador del lenguaje de definición de datos.
- **Concesión de autorización para el acceso a los datos:** esto para regular qué partes de la BDD van a poder ser accedidas por varios usuarios.
- **Especificación de las restricciones de integridad:** las restricciones se mantienen en una estructura especial del sistema que consulta el gestor de la BDD cada vez que tiene lugar una actualización en el sistema.

• SISTEMA DE BASE DE DATOS

Es un sistema de mantenimiento de registros basado en computadores, cuyo propósito general es registrar y mantener los datos, tales datos pueden estar relacionados con cualquier cosa que sea significativa para la organización donde el sistema opera, en otras palabras cualquier dato necesario para los procesos de toma de decisiones inherentes a la administración de esa organización.

Un sistema de base de datos incluye cuatro componentes principales: datos e información, software, hardware y usuarios.

Datos.— los datos almacenados en el sistema se dividen en una o más tablas o record types. Desde este punto

de vista existe una base de datos, la que contiene todos los datos almacenados en el sistema.

Una base de datos es un repositorio de datos almacenados los que son **integrados** como **compartidos**.

Por **INTEGRADA** se entiende que la base de datos puede considerarse como una unificación de varios archivos de datos independientes, donde se elimina parcial o totalmente cualquier redundancia entre los mismos.

Por **COMPARTIDA** se entiende que partes individuales de la base de datos pueden compartirse entre varios usuarios distintos en el sentido de que cada uno de ellos puede tener acceso a la misma base de datos y utilizarla con propósitos diferentes. Tal compartimiento es una verdadera consecuencia del hecho de que la base de datos es integrada.

La palabra compartida a menudo se amplía para abarcar no sólo lo antes descrito, sino también al compartimiento concurrente, es decir la oportunidad que diversos usuarios accedan en realidad la base de datos al mismo tiempo. Un sistema de base de datos que admite esta forma de compartimiento se llama sistema de usuarios múltiples.

HARDWARE

El hardware se compone de los volúmenes de almacenamiento secundario como cintas o discos, donde reside la base de datos, junto con dispositivos asociados como las unidades de control, los canales, etc. La base de datos es demasiado grande para caber en su totalidad en la memoria principal del computador.

SOFTWARE

Sistemas operativos

Entre la base de datos física en sí, es decir, el almacenamiento real de los datos y los usuarios del sistema existe un nivel de software que a menudo recibe el nombre de sistema de administración de bases de datos o **DBMS**. Este maneja todas las solicitudes de acceso a la base de datos, solicitadas por los usuarios. Una función general del DBMS es proteger a los usuarios de la base de datos contra los detalles a nivel de hardware, casi de la misma manera en que los sistemas de lenguajes de programación para lenguajes como COBOL protegen a los usuarios programadores en los detalles a nivel de hardware. En otras palabras el DBMS ofrece una vista de la base de datos que está por encima del nivel de hardware y apoya las operaciones del usuario.

El conjunto de datos se conoce como base de datos, ésta contiene información acerca de una empresa determinada, el objeto primordial de un DBMS es crear un ambiente en el que sea posible guardar y recuperar datos e información de la base de datos en forma conveniente y eficiente.

Los sistemas de base de datos se diseñan para manejar grandes cantidades de datos e información, el manejo de los datos incluye toda la definición de las estructuras para el almacenamiento de los datos como los mecanismos para el manejo de la información, así mismo el sistema de base de datos deben cuidar la seguridad de la información almacenada en la base de datos tanto contra las caídas del sistema como contra los intentos de acceso no autorizado. Si los datos van a ser compartidos por varios usuarios, el sistema debe evitar la posibilidad de obtener resultados análogos.

USUARIOS

Se consideran tres grupos de usuarios:

La primera la representa el programador de aplicaciones, encargado de escribir programas de aplicación que utilice la base de datos. Estos programas de aplicación operan con los datos de todas las maneras usuales; recuperan información, crean nueva información, suprimen o cambian informaciones existentes. Todas estas funciones se realizan formulando las solicitudes adecuadas al DBMS. Los programas en sí pueden ser aplicaciones convencionales de procesamiento por lotes o programas en línea diseñados para apoyar al usuario final, que interactúa con el sistema desde una terminal en línea.

La segunda clase de usuario es el usuario final que accesa la base de datos desde una terminal. Un usuario final puede emplear un lenguaje de consulta (SQL) proporcionado como parte integral del sistema o recurrir a un programa de aplicación escrito por un usuario programador que acepte órdenes desde la terminal y a la vez formule solicitudes al DBMS en nombre del usuario final. El usuario final puede realizar, en general, las funciones de recuperación, creación, supresión y modificación.

La tercer clase de usuario la representa el administrador de bases de datos o **DBA** (Data Base Administrator). Las funciones del administrador de base de datos son entre otras las siguientes:

1. Definición del Esquema.– la creación del esquema original de la base de datos. Esto se logra escribiendo una serie de definiciones que el compilador del Lenguaje de definición de datos (DDL) traduce a un conjunto de tablas que se almacenan permanentemente en el diccionario de datos.

ESQUEMAS.– Se llama esquema a la descripción lógica de la Base de Datos; El esquema es un diagrama de los tipos de datos que se usan, y proporciona los nombres de las entidades y sus atributos, además especifican las relaciones que existe entre ellos. (El esquema permanece, los valores pueden variar).

2. Definición de la estructura de almacenamiento y del método de acceso.– la creación de de las estructuras y almacenamiento y de métodos de acceso apropiados. Esto se lleva a cabo escribiendo una serie de definiciones que posteriormente son traducidas por el compilador del lenguaje de almacenamiento y definición de datos.

Conceptos básicos de SQL

Sentencias:

El lenguaje SQL consta de treinta sentencias, cada sentencia demanda una acción específica por parte del DBMS, tal como la creación de un anueva tabla, la recuperación de datos o la inserción de nuevos datos en la base. Todas las sentencias SQL tienen la misma forma básica.

Sentencia Descripción

Manipulación de datos

SELECT Recuperar datos de la base de datos

INSERT Añade nuevas filas de datos a la base de datos

DELETE Suprime filas de datos a la base de datos

UPDATE Modifica datos existentes en la base de datos

Definición de datos

CREATE TABLE Añade una nueva tabla a la base de datos

DROP TABLE* Suprime una tabla de la base de datos

ALTER TABLE* Modifica de una tabla existente

CREATE VIEW* Añade una nueva vista a la base de datos

DROP VIEW Suprime una lista de la base de datos

CREATE INDEX* Construye un índice para una columna

DROP INDEX* Suprime el índice para una columna

CREATE SYNONYM* Define un alias para un nombre de tabla

DROP SYNONYM* Suprime un alias para un nombre de tabla

COMMENT* Define comentarios para una tabla

LABEL Define un título de una columna

Control de acceso

GRANT Concede privilegios de acceso a usuarios

REVOKE Suprime privilegios de acceso a usuarios

Control de transacciones

COMMIT Finaliza la transacción actual

ROLLBACK Aborta la transacción actual

SQL programático

DECLARE Define un cursor para una consulta

EXPLAIN* Describe el plan de acceso a datos para una consulta

OPEN Abre un cursor para recuperar resultados de consulta

FETCH Recupera una fila de resultados de consulta

CLOSE Cierra un cursor

PREPARE* Prepara una sentencia SQL para ejecución dinámica

EXECUTE* Ejecuta dinámicamente una sentencia SQL

DESCRIBE* Describe una consulta preparada

* No forma parte del estándar SQL ANSI/ISO, pero se encuentra en la mayoría de los productos más populares basados en SQL.

Todas las sentencias SQL empiezan con un **verbo**, una palabra clave que describe lo que la sentencia hace. CREATE, INSERT, DELETE y COMMIT son verbos típicos. La sentencia continúa con una o más **cláusulas**. Una cláusula puede especificar los datos sobre los que debe actuar la sentencia, o proporcionar más detalles acerca de lo que la sentencia se supone que hace.

3. Modificación del esquema y de la organización física.– el DBA puede realizar la modificación del esquema de la base de datos o de la descripción de la organización física del almacenamiento. Estos cambios aunque son relativamente poco comunes, se logran escribiendo una serie de definiciones utilizadas, ya sea por el compilador del DDL o por el compilador del lenguaje de almacenamiento y definición de datos para generar modificaciones a las tablas internas apropiadas del sistema.

Un sistema de base de datos proporciona a la empresa un control centralizado de sus datos de operación lo que implica, que en una empresa que utilice un sistema de base de datos debe existir una persona específica cuya responsabilidad central sea controlar los datos de operación, ésta persona es el administrador de las bases de datos DBA.

Las ventajas de tener un control centralizado de los datos son:

Puede reducirse la redundancia

En sistemas que no se usa base de datos cada aplicación tiene sus propios archivos, esto origina enormes redundancias en los datos almacenados y como resultado desperdicio del espacio de almacenamiento, en un sistema de base de datos la redundancia debe controlarse, es decir el sistema debe estar al tanto y asumir las responsabilidades de propagar las actualizaciones.

Puede evitarse la inconsistencia

Esto en realidad es corolario (similar) del punto anterior, es decir cuando existen distintas copias de los mismos datos y éstos no concuerdan entre sí.

Los datos pueden compartirse

No sólo significa que las aplicaciones existentes puedan compartir los datos de la base de datos, sino también que es factible desarrollar nuevas aplicaciones que operen con los datos almacenados.

Pueden hacerse cumplir las normas establecidas.

Con un control central de la base de datos el DBA puede garantizar que se cumplan todas las formas aplicables a la representación de los datos, las normas aplicables pueden comprender la totalidad o parte de lo siguiente: Normas de la compañía, de instalación, departamentales, industriales, etc. es muy deseable unificar los formatos de los datos almacenados como ayuda para el intercambio o migración de datos entre sistemas.

Pueden aplicarse restricciones de seguridad.

Al tener jurisdicción sobre los datos de operación el DBA puede:

- Asegurar que el único medio de acceder la base de datos sea a través de los canales establecidos
- Definir controles de automatización para que se apliquen cada vez que se intente el acceso de datos. Los

controles pueden establecerse para cada tipo de acceso: recuperación, modificación, eliminación, a cada parte de la información de la base de datos

Puede conservarse la integridad.

El problema de la integridad es garantizar que los datos de la base de datos sean exactos. La inconsistencia entre dos entradas que representan el mismo hecho es un ejemplo de falta de integridad que por supuesto sólo ocurre si existe redundancia de los datos almacenados aún cuando la redundancia se elimine la base de datos puede contener aún datos incorrectos.

El control centralizado de la base de datos ayuda a evitar éstas situaciones en la medida de lo posible pues permite al DBA definir procedimientos de validación que hay que ejecutarse cada vez que se intente una actualización, es conveniente señalar que la integridad de los datos es más importante en un sistema de bases de datos que en un sistema de archivos privados.

Pueden equilibrarse los requerimientos contradictorios.

Cuando conoce los requerimientos globales de la empresa puede el DA estructurar el sistema de base de datos para brindar un servicio que sea el mejor para la empresa en términos globales por ejemplo puede elegirse una representación de los datos almacenados que ofrezca rápido acceso a las aplicaciones más importantes a costa de un desempeño de menor calidad en algunas otras aplicaciones.

Arquitectura de una Base de Datos.– La arquitectura se divide en tres niveles generales: interno, conceptual y externo.

El nivel interno es el más cercano al almacenamiento físico, es decir es el que contiene a la manera como los datos se almacenan en realidad.

El nivel externo es el más cercano a los usuarios, es decir el que atañe a la manera cómo cada usuario ve los datos.

El nivel conceptual es la planificación del problema a resolver, es dónde se examinan las entidades, los atributos y el sondeo de los registros a resolver.

En un sistema de base de datos existirá muchas vistas externas, cada una compuesta por una una representación más o menos abstracta de alguna parte de la base de datos, y habrá una sola vista conceptual, compuesta por una representación más o menos abstracta de la base de datos en su totalidad. Asimismo habrá una sola vista interna que representa la base total como esta almacenada.

EL ADMINISTRADOR DE LA BASE DE DATOS

Es importante considerar las actividades destinadas a introducir los datos en la base, a controlar lo que ahí se encuentra almacenado y a proporcionar información a partir de ellos. Para describir estas actividades se emplean muchas expresiones como por ejemplo: actualización y recuperación, administración de archivos, manejo de datos, administración de bases de datos, etc. Aunque estas expresiones se refieren a las mismas actividades principales, existen algunas diferencias significativas entre ellas.

EL ENFOQUE DE APLICACION

Tradicionalmente los archivos de datos se diseñaron para responder a ciertas aplicaciones individuales, por

ejemplo la elaboración de facturas y nóminas, control de cuentas por cobrar e inventarios , etc. Cada archivo de datos se diseño con su propia área de almacenamiento, ya sea dentro del computador o en dispositivos auxiliares como lo son la cinta o el disco magnético.

En primer lugar es preciso formular uno o más programas para actualizar cada uno de los archivos maestros que contienen datos de las transacciones.

En segundo se necesita uno o más programas para sustituir o modificar los archivos maestros en los cuales se han encontrado errores.

En tercer lugar, cada vez que se requiere un determinado informe se necesita uno o más programas.

En cuarto lugar, para relacionar dos o más archivos maestros se requiere de uno o más programas especiales, en general, este enfoque esta orientado hacia el procesamiento por lotes.

En muchas empresas los archivos de datos diseñados oriegnalmente en forma independiente, o archivos únicos, estan de hecho vinculados con estructuras complejas de programción. En otras los archivos se incluyen en relaciones específicas y rígidas mediante una amplia programación. En los dos casos una buena parte del costo de desarrollo se invierte en adaptaar los programas para llevar a cabo las actividades de administración de las bases de datos. Por otra parte cada vez que se modifica un archivo o se altera la necesidad de cierto informe, se requiere una buena inversión para modificar el programa adaptado.

El enfoque de la administración de archivos

Los primeros intentos de minimizar y resolver la administración de las bases de datos estuvieron dirigidos hacia los archivos individuales y a la obtención de datos de dichos archivos. Los sistemas de manejo de archivos que el programador indicará únicamente el formato del archivo y de la información requerida. Una vez dados estos parámetros, el programa generalizado consultaba el archivo y extraía los elementos necesarios, efectuaba algunas operaciones lógicas y matemáticas limitadas y producía la información.

Al continuarse los esfuerzos para mejorar los sistemas de administración de archivos, se lograron las siguientes ventajas:

1. La posibilidad de procesar más de un archivo.
2. La posibilidad de procesar archivos organizados de forma no secuencial.
3. Mayor capacidad lógica y matemática.
4. Capacidad para más de una forma de producción informativa.
5. Posibilidad de procesar con dispositivos interactuantes, como pantallas y teletipos.

Las expresiones "sistema de administración de datos" y sistemas de administración de base de datos" comenzaron a relegar al "Sistema de administración de archivos" por:

1. La posibilidad de actualizar y conservar
2. La posibilidad de tener acceso a un mismo archivo con dos o más programas simultaneamente.
3. La separación del formato del archivo con respecto a los programas de aplicación.

ENFOQUE GENERALIZADO DE LA ADMINISTRACION DE BASE DE DATOS

Los fines principales del enfoque generalizado de la administración de la base de datos son las siguientes:

1. Obtener independencia de los datos.
2. Disminuir la duplicación de los datos.
3. Relacionar lógicamente los datos y hacer que sean consistentes con los aspectos funcionales de los usuarios.
4. Reducir los costos de aplicación, almacenamiento y procesamiento.

La independencia de los datos ofrece la posibilidad para utilizarlos para más de un programa simultáneamente. Asimismo permite introducir un cambio en la definición de los datos sin el correspondiente cambio en los programas. La independización de los datos ocasiona una reducción en el costo del programa de aplicación.

PROCESO DE DISEÑO DE UNA BASE DE DATOS

METODOLOGIA DEL DISEÑO

Un método es un proceso ordinario y lógico para realizar una tarea. Una metodología es un sistema de métodos, que aplica los principios básicos del razonamiento. En términos de base de datos, la metodología del diseño puede ser pensada como la colección de herramientas y técnicas empleadas en la organización de una aplicación, que puede ser aplicada constantemente en la creación de estructuras para las bases de datos en incontables proyectos. Una base de datos se constituye tanto de programas como de datos, la metodología de diseño de base de datos se considera como una parte integral.

Las metas de una buena metodología de diseño de datos son:

1. Debería producirse una estructura útil en un tiempo y esfuerzo razonable. Una base de datos útil es aquella en donde se encuentran los objetivos del usuario ejemplo: alta eficiencia, adaptabilidad para futuras modificaciones, seguridad, integridad, etc.
2. Una metodología deberá ser lo bastante general y flexible para ser no solamente usada por personas de experiencia en el diseño.
3. Una metodología debería ser reproducible así que dos personas o programas que apliquen igual metodología al mismo problema producirá la misma o aproximadamente el mismo resultado.

La metodología de diseño de base de datos se presenta en el siguiente gráfico.

Información general de Procesamiento de requerimientos Requerimientos

%%%%%%%%%

%%%%%%%% %

% %%%%%%%%%%

% % Paso 1 %

% % Formulación de requerimientos %

% % y Análisis de Sistemas %

% %%%%%%%%%%

% Especificacion % de requerimientos % %%%Carat.

% % % %de la

% %%%%%%%%%% % %Base de

%%%%%%%%% Paso 2 % % %Datos

% Diseño Conceptual % % %(DBMS)

%%%%%%%%%%

% % %

Estructura de %la información %

% % %

%%%%%%%%%%

% Paso 3 %%%%%%%%%

% Diseño de Implementación %%% %

%%%%%%%%%%

Estructura lógi- %ca de la Base de %

Datos y especi - %ciones para pro- %

gramas % % %

% % %

%%%%%%%%%%

% Paso 4 %%%%%%%%%

% Diseño Físico %%%Carac.

%%del hardware

y Sis.Operat.

Base de Datos (database)

Se puede definir como una colección de datos que satisface las necesidades de multiples usuarios con una o más organizaciones. Un punto destacable de una base de datos es el ser una fuente integrada, que puede ser usada por todos los miembros de una organización quienes necesitan información contenida en esta base (base de datos).

Una base de datos debería soportar:

1. Rapidez (speed) El sistema automatizado permite consultas en línea (on line queries).
2. Disponibilidad (total availability) Toda la información contenida en una base de datos debe estar disponible para su uso.
3. Flexibilidad (flexibility) Que los cambios sean relativamente fáciles de implementar.
4. Integridad (integrity) La duplicación de datos debe ser en lo posible reducida y las políticas de modificación deben ser estandarizadas, resultando una base de datos consistente.

Niveles de representación de datos

Tres niveles de abstracción de datos pueden ser reconocidos para poder especificar la estructura de una base de datos: La conceptual o vista del empresario, la vista de la implementación del programador de aplicaciones y la vista física

NIVEL CONCEPTUAL -----> Vista del empresario

- Entidades (Entities)
- Atributos (Attributes)
- Relaciones (Relationships)

NIVEL DE IMPLEMENTACION -----> Vista del programador

- Registros (Records)
- Campos (Data Items)
- Sets (Interrecord Relationships)

NIVEL FISICO

- Bloques (block)
- Punteros (Pointers)
- Controles (Overhead data)
- Clusterings

Vista Conceptual o Nivel conceptual.– Es esquema conceptual o la estructura de la información es el nombre dado a la estructura de datos en el nivel conceptual, es la orientación de problema, del mundo real y es completamente independiente del sistema DBMS, del Sistema Operativo y del Computador.

La Estructura conceptual, o esquema (schema) consiste de elementos básicos del mundo real (personas o cosas) llamadas entidades (entities); otros elementos los cuales describen las entidades, llamados atributos, y las asociaciones entre las ocurrencias de los elementos básicos entre entidades llamadas relaciones (relationships). Cabe señalar que existen tres tipos de relaciones binarias entre las ocurrencias de dos entidades:

* Uno a Uno (One – to – One) 1:1

* Uno a muchos (One – to – Many) 1:n

* Muchos a muchos (Many – to – Many) n: m

Las relaciones son constituídas desde N-Tuples. Un N-tuple es una colección de N valores de datos, dónde cada valor ha sido seleccionado desde un dominio particular. Los valores de datos pueden ser seleccionados desde un común dominio o distintos dominios. Por ejemplo:

<LA FAVORITA, A123, Loja)

Es un 3-tuple de valores de los dominios, NOMBRE_FABRICA, NUMERO DEL PRODUCTO y LOCALIZACION DE LA FABRICA.

Una relación es un conjunto de tuples, colocados en una tabla en la que los **tuples** forman las **filas** y los **dominios** las columnas. Es significativo hacer notar que "SET" es usado aquí en sentido matemático, los tuples en una relación no aparecen en cualquier secuencia y las relaciones no pueden tener tuples duplicados.

Normalización

Es hacer posible la tabulación o el ingreso de cualquier relación en la base de datos o en archivo de datos.

Con la normalización se puede conseguir una capacidad de recuperación de datos potente mediante una serie simple de operaciones relacionales.

El Modelo de Datos Relacional especifica que las relaciones esten en la tercer forma normal.

Primer forma normal.– todos los dominios deben contener solamente datos básicos (valores básicos) o es la representación de los datos en forma natural. Sirve de base para un lenguaje de alto nivel que proporcione la máxima independencia de datos.

Segunda forma normal.– cada dato que no es llave debe ser funcionalmente dependiente sobre la primer llave o todo atributo que no es clave es totalmente y funcionalmente dependiente sobre la llave primaria.

Tercer forma normal.– cada dato que no es llave no puede ser funcionalmente dependiente sobre cualquier otro dato no llave del dominio o todo atributo que no es clave no es transitivamente dependiente sobre la clave primaria.

GLOSARIO UTILIZADO EN LA DEFINICION DEL DICCIONARIO DE DATOS

Siendo el diccionario de datos la herramienta de comunicación, está debería ser efectiva y debería existir un entendimiento entre los usuarios acerca de su construcción. Las siguientes son las partes básicas de un diccionario de base de datos

Elementos de datos.– es un atributo que describe una entidad. Todos los elementos de datos tienen un nombre único, el nombre o rótulo tiene que ser descriptivo y deberá consistir de un conjunto de palabras o abreviaturas tomadas de una lista aprobada.

Grupo de elementos.– es un grupo de atributos logicamente relacionados, para un grupo de atributos todos los subelementos , deben ser explícitamente identificados.

Elementos de datos derivados.– es un elemento de dato que se origina como resultado de una fórmula o cálculo. Para todo elemento de dato derivado, la fórmula o cálculo debe ser identificado, a lo largo con todos los elementos de datos usados en la derivación. Estos elementos de datos solamente deberán ser definidos en el diccionario.

Sinónimos o Sobrenombres.– es un elemento de dato con diferente designación pero con el mismo significado como otro elemento de datos la descripción de un sinónimo debe identificar al designador de otro elemento de dato con el cual este es sinónimo.

Antónimo.– son dos elementos de datos con la misma designación pero con diferente significado. Los problemas de sinónimo de sinónimo y antónimo deben ser identificados por el diccionario y los problemas deben ser resueltos por el equipo de diseño de la base de datos.

Descripción del modelo conceptual.– el modelo describe las entidades de la empresa, las relaciones entre las entidades y el flujo de información.

Descripción del modelo lógico.– el modelo es posiblemente un subconjunto del modelo conceptual y es mapeado (tiene iguales características) al sistema de datos a ser usado.

Descripción del modelo externo.– los modelos externos describen la vista de los programas de aplicación que son los caminos por los cuales las relaciones entre las entidades son accedidas por diferentes aplicaciones.

Descripción del modelo interno.– El modelo interno es el mapa físico de la base de datos. Describe las relaciones entre las entidades, los métodos de acceso usados, factores de bloqueo y otros detalles físicos.

DICCIONARIO DE DATOS

El primer paso en el diseño de una base de datos es recolectar información acerca de la empresa, que es,

acerca de su uso, relaciones y significado de datos. Como el diseño de procesos es progresivo, es necesario depositar información acerca de la opinión lógica, interna y externa del modelo en la localización central. Una herramienta que facilita el control y manejo de la información acerca de datos en el diseño, implementación, operación y expansión de fases de una base de datos es llamado **diccionario de datos**.

El diccionario de datos es un lugar dónde se deposita información acerca de datos como origen, descripción, relaciones y otros datos, es decir el diccionario de datos es una base de datos misma, la cual deposita datos acerca de datos, el diccionario de datos es una guía y contiene "mapas guías" para la base de datos en vez de "nuevos datos", es decir es un lugar en dónde se almacena o se mantiene un conjunto de estados (controles), información relacionada con los diferentes tipos de registros (tablas) privilegios de los usuarios y estadísticas (cuantos registros tiene cada tabla, índices, etc.)

Los diccionarios de datos de los Sistemas de Base de datos (DBMS) no son iguales, aunque mantienen los mismos lineamientos o las mismas características.

DICCIONARIO DE DATOS Y LAS INTERFASES

El diccionario de datos puede componerse básicamente de dos interfases así:

1. La interfase con la gente involucrada, por ejemplo, el administrador de la base de datos, programador de sistemas, programador de aplicaciones, manejadores, y finalmente usuarios y observadores.
2. La interfase con el software por ejemplo, sistema de manejo de bases de datos, librerías, sistemas operativos y generador de reportes.

El diccionario de datos puede ser usado como una herramienta efectiva para la función de administrador de base de datos en el diseño, implementación y fase de operaciones en la base de datos. Es responsabilidad del DBMS proteger el diccionario de datos por refuerzos estándar, seguridad y obligaciones privadas. Un diccionario de datos es el lugar ideal para encontrar respuestas a las preguntas como "**dónde se usa**", "**quién usa**", "**cuando se usa**".

Estas interfases muestran que existen dos tipos de usos del diccionario de datos, un tipo de uso es por la gente que tiene funciones como administrador de base de datos, programador de sistemas, analista de sistemas, programador de aplicaciones, usuarios. Y el otro tipo de uso es por el software en áreas semejantes como manejadores de base de datos, sistemas, librerías, sistemas operativos y generador de reportes.

Estos dos tipos de interfase enlazan al manejador y control del medio de la base de datos como un resultado de la eficiente comunicación entre las partes involucradas.

DICCIONARIO DE DATOS IDEAL: SUS REQUERIMIENTOS

Y SU ORGANIZACION

La siguiente es una lista de requerimientos convenientes de un diccionario de base de datos para describir los datos, no quiere decir que cualquier paquete de diccionario de datos particular abarca ahora todos estos requerimientos.

Modelo Conceptual.— la información acerca de los datos necesaria en el proceso de diseño del modelo conceptual incluye entidades, campos o atributos y las relaciones entre campos, atributos, también incluye información acerca de cuales departamentos y usuarios estan usando o intentan usar que atributos y con que

frecuencia estos datos son usados, conjuntamente con las descripciones textuales y con significados y propósitos. Las entidades y relaciones deberían tener títulos apropiados, versiones, estados, los membership (campo de una entidad el cual va a servir de referencia).

Modelo Lógico.— la información siguiente acerca del modelo lógico de la base de datos debería ser almacenada en el diccionario de datos: el campo de agrupación con su llave (estos grupos pueden ser los subgrupos de los grupos especificados en el modelo conceptual), el fundamento del modelo de datos, la relaciones de los grupos basados en el modelo de datos, el modelo externo soportado por el modelo lógico, las transacciones lógicas, los programas y los módulos, la referencia cruzada de la información entre las transacciones, también deberían ser almacenados. Otra información necesaria es el lenguaje de programación y el tipo de programa (batch o en línea) para los programas y transacciones.

Modelo Interno.— la información física acerca de los atributos como por ejemplo: longitud (caracteres), modo (cadena de caracteres, decimales, datos de simple precisión, empaquetados), justificación (derecha izquierda), formas de presentación, reglas de edición (constantes, rango de valores), derivación algorítmica, secuencia o posición secuencial que un atributo particular ocupa en una ocurrencia, seguridad (códigos de seguridad para leer, actualizar), medio de almacenamiento (tarjetas, discos, cintas, video), el control de acceso a la información debería ser almacenado en el diccionario de datos.

Un diccionario de datos ideal debería ser una parte integral de todo el medio ambiente de la base de datos y el administrador de la base de datos es el responsable de la entrada al diccionario de datos, señalando que un diccionario de datos tiene que ser salvado en copias de respaldo para evitar efectos desastrosos debido a un mal funcionamiento del sistema o cualquier destrucción no intencional de la versión producida del diccionario de datos, la función del administrador de base de datos lleva la gran responsabilidad de proteger la parte vital del medio de la base de datos "el diccionario de datos".

Sistemas Ideales del diccionario de datos

1. El diccionario de datos debe soportar los modelos conceptual, lógico, interno y externo.
2. El diccionario de datos debe ser integrado con el manejador del sistema de base de datos.
3. El diccionario de datos debe soportar varias versiones de documentación (historial)
4. El diccionario de datos debe apoyar la transferencia eficiente de información al manejador del sistema de base de datos. Idealmente la conexión entre los modelos interno y externo debe ser realizada en tiempo de ejecución.
5. Un diccionario de datos debería comenzar con la reorganización de versiones de producción de la base de datos como un resultado de los cambios para la descripción de la base de datos. Similarmente, cualquier cambio a la descripción de programas debe ser reflejado automáticamente en la librería de descripción de programas con la ayuda del diccionario de datos.
6. El diccionario de datos para ser eficiente deberá ser almacenado en un medio de almacenamiento con acceso directo para la fácil recuperación de información.

Para llegar al modelaje entidad/relación se recomienda seguir los siguientes pasos:

1. IDENTIFICAR LAS ENTIDADES

Se debe identificar todos aquellos objetos reales o abstractos que son significativos para el sistema en desarrollo.

2. IDENTIFICAR RELACIONES

Una relación es cualquier tipo de asociación entre las entidades mediante uno o varios atributos comunes, siempre y cuando dicha asociación sea relevante y estable para el sistema en estudio.

3. REALIZAR EL DIAGRAMA ENTIDAD/RELACION

Es de gran ayuda por cuanto nos permite tener una representación gráfica del flujo de datos del sistema en desarrollo, asimismo nos sirve como documentación para efectos de validación y verificación con el usuario final. El diagrama E-R que consta de los siguientes componentes:

- * Rectángulo representan conjuntos de entidades
- * Círculos representan atributos
- * Rombos representan relaciones entre conjuntos de entidades
- * Líneas que conectan los atributos a los conjuntos de entidades y los conjuntos de entidades a las relaciones

Cada componente se etiqueta con el nombre correspondiente.

Ejemplo puntualizando:

Para realizar el análisis de un problema se debe:

1. Definir el problema.– conocer el objetivo y cumplimiento de las tareas
2. Análisis del Problema.– recoger, organizar e interpretar los datos que tienen relación con el problema

Arquitectura.– el sistema relacional se compone de dos subsistemas principales:

- Sistema de Almacenamiento e Indagación o **RSS**
- Sistema de Datos Relacional o **RDS**

El Sistema de datos relacional **RDS** proporciona la interface al **usuario externo**, que soporta las estructuras de datos tabulares y los operadores sobre esas estructuras y el **RSS** o sistema de almacenamiento e indagación, proporciona al Sistema de datos relacional una interface de **registros almacenados**.

Sistema de almacenamiento de indagación (RSS)

El RSS es ante todo un poderoso método de acceso. Su función principal consiste en manejar todos los detalles de nivel físico y ofrecer a su usuario una interface llamada interface de Almacenamiento e Indagación o RSI, sin embargo el usuario de RSS no siempre es un usuario directo, sino el código generado por el sistema de datos relacional en la compilación de alguna proposición de SQL.

El objeto de datos básico soportado en la RSI es el archivo almacenado, es decir, la representación interna de

una tabla de base. Los renglones de la tabla se representan por medio de registros del archivo, sin embargo los registros almacenados dentro de un archivo almacenado no necesitan hallarse en contigüidad física en el almacenamiento. El RSS admite también un número arbitrario de índices sobre cualquier archivo almacenado. Se proveen operadores en la RSI para buscar en todo un archivo almacenado en una secuencia del sistema (definida por RSS) y en una secuencia acorde con cualquier índice especificado. El usuario de la RSI necesita saber qué archivos almacenados e índices existen, y debe especificar la ruta de acceso que ha de usarse en cualquier solicitud dada de acceso de RSI.

Sistema de Datos Relacionales (RDS)

El RDS, a su vez está integrado por dos componentes: un precompilador y un sistema de control a tiempo de ejecución.

El **precompilador** es un compilador para el lenguaje SQL. Supóngase que el programador de aplicaciones escribe un programa P que incluye algunas proposiciones inmersas de SQL. Supongamos que P se escribe en COBOL, antes que P pueda ser compilado de la manera usual por el compilador de COBOL, primero debe ser procesado por el precompilador de RDS. La precompilación opera de la siguiente manera:

1. El precompilador recorre el programa fuente P y localiza las proposiciones de SQL inmersas;
2. Para cada proposición que halla, el precompilador opta por una estrategia para realizar esa proposición en términos de las operaciones de RSI, este proceso se llama optimización. La optimización se fundamenta en el conocimiento del precompilador de las rutas de acceso disponibles en la RSI. Después de tomar sus decisiones, el precompilador genera una rutina en lenguaje de máquina que realizará la estrategia elegida. El conjunto de todas estas rutinas constituye el módulo de acceso para el programa P específico. El módulo de acceso en sí se almacena en la base de datos.
3. El precompilador reemplaza a cada una de las proposiciones inmersas originales de SQL por una proposición ordinaria de COBOL para llamar a XRDI (XRDI es el nombre del componente de control a tiempo de ejecución de RDS). El Sistema de control de tiempo de ejecución (XRDI) suministra el ambiente de ejecución para un programa de aplicación que haya pasado por el proceso de precompilación

El programa fuente modificado cobol puede ahora compilarse de manera normal mediante el compilador de cobol. Para el caso del lenguaje en línea (presentado por medio de la interface amigable con el usuario) el proceso en esencia es similar, excepto que la ejecución sucede tan pronto como la precompilación se completa.