

BASES DE DATOS
TEMA 3
MODELO ENTIDAD - RELACIÓN

3.3 Aplicaciones

Definición de Aplicación (Application). Programa informático que permite a un usuario utilizar una computadora con un fin específico. Las aplicaciones son parte del software de una computadora, y suelen ejecutarse sobre el sistema operativo.

Una aplicación de software suele tener un único objetivo: navegar en la web, revisar correo, explorar el disco duro, editar textos, jugar (un juego es un tipo de aplicación), etc. Una aplicación que posee múltiples programas se considera un paquete. Son ejemplos de aplicaciones Internet Explorer, Outlook, Word, Excel, Dreamweaver, etc.

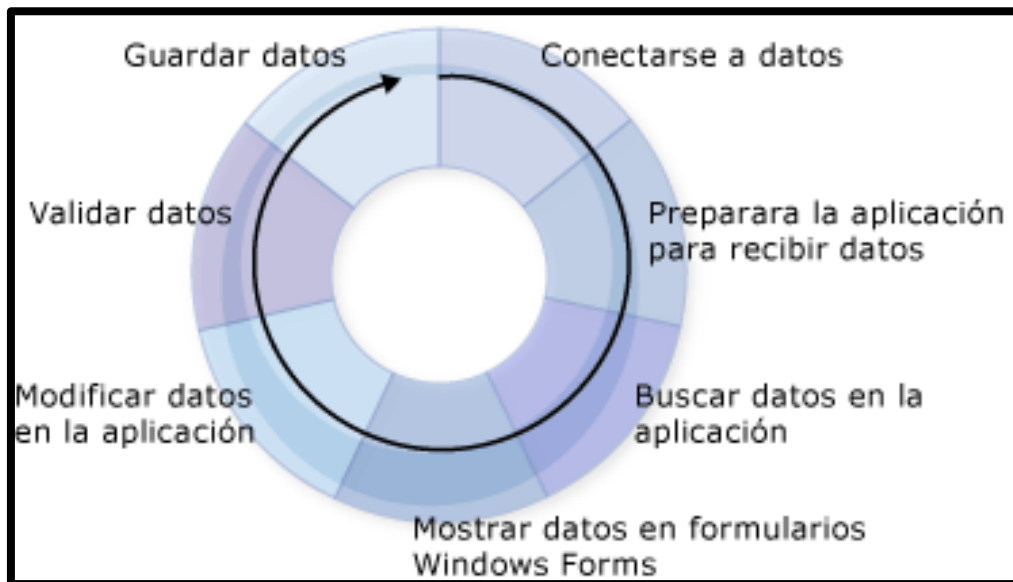
En general, una aplicación es un programa compilado (aunque a veces interpretado), escrito en cualquier lenguaje de programación.

Las aplicaciones pueden tener distintas licencias de distribución como ser freeware, shareware, trialware, etc. Las aplicaciones tienen algún tipo de interfaz, que puede ser una interfaz de texto o una interfaz gráfica (o ambas).

Debido a la creciente aceptación de las bases de datos por parte de la industria y el gobierno en el plano comercial, y a una variedad de aplicaciones científicas y técnicas, el diseño de bases de datos desempeña un papel central en el empleo de los recursos de información en la mayoría de las organizaciones.

Una aplicación de datos típica utiliza la mayoría de los procesos que se ilustran en el diagrama siguiente:

El ciclo de datos



Conectarse a datos

Para llevar datos a la aplicación (y devolver los cambios al origen de datos), debe establecerse algún tipo de comunicación bidireccional. Esta comunicación bidireccional la controlan, por lo general, los objetos de su modelo de datos.

Preparara la aplicación para recibir datos

Si la aplicación usa un modelo de datos desconectado, necesita almacenar temporalmente los datos en la aplicación mientras trabaja con ella.

Buscar datos en la aplicación

Independientemente de que la aplicación utilice un modelo de datos desconectado, necesitará obtener datos. Lleve los datos a la aplicación mediante la ejecución de consultas o procedimientos almacenados de una base de datos. Las aplicaciones que almacenan datos en conjuntos de datos ejecutan las consultas y los procedimientos almacenados mediante objetos mientras que las aplicaciones que almacenan datos en entidades ejecutan las consultas conectando las entidades directamente con los procedimientos almacenados.

Mostrar datos en formularios

Después de introducir los datos en la aplicación, los mostrará en un formulario para que los usuarios los vean o los modifiquen.

Modificar datos en la aplicación

Cuando ya ha presentado los datos a los usuarios, es probable que los modifiquen, agregando, cambiando o eliminando registros antes de devolverlos a la base de datos.

Validar datos

Al realizar cambios en los datos, se necesita comprobar los cambios antes de permitir que los valores sean aceptados de regreso en la base de datos o que éstos se escriban en ella. Validación es el nombre del proceso que comprueba que estos nuevos valores son aceptables para los requisitos de la aplicación. Puede agregar lógica para comprobar los valores de la aplicación a medida que se modifican.

Guardar datos

Después de realizar los cambios en la aplicación (y validarlos), es necesario enviar los cambios a la base de datos.

Los programas de bases de datos organizan y almacenan los datos de manera tal que las tablas están indexadas y se pueden contestar preguntas. Estas herramientas están disponibles en un número de configuraciones diferentes y escalables para ser usadas por un individuo o una corporación global. Algunos son fáciles de usar, con bases de datos a manera de interfaz visual, mientras que otras, al final del espectro empresarial, requieren entrenamiento y herramientas especializadas para ser útiles. Algunos ejemplos de estas aplicaciones son las siguientes:

- **Access:** Es un sistema de base de datos personal de Microsoft. Se trata de un producto de software orientado hacia lo visual, lo que hace que quienes no sean programadores puedan crear bases de datos útiles con facilidad. Su uso más común es para pequeñas bases de datos individuales o en programas multiusuario de uso limitado. Access integra el lenguaje Visual Basic para aplicaciones.
- **Visual FoxPro:** Es un sistema de base de datos relacional, también producido por Microsoft, que está estrechamente unido a su lenguaje de programación. Este

sistema de base de datos es conocido por su motor de procesamiento rápido y la capacidad de manejar numerosas transacciones simultáneas.

- **MySQL Database:** Es una base de datos basada en servidor que permite a varios usuarios acceder a múltiples bases de datos. El software funciona en múltiples plataformas, incluyendo la mayoría de las variedades de UNIX y Windows. Ofrece usabilidad de primer plano limitada y está diseñado como un servidor de base de datos back-end. MySQL se diferencia de otros productos de base de datos por sus costos, la versión no empresarial se distribuye de forma gratuita.
- **SQL Server:** Es un servidor de base de datos a nivel empresarial escalable. Este producto se diferencia de la base de datos personal al no proporcionar las herramientas para el usuario que proporciona un producto de base de datos individual. El motor de base de datos se centra en responder rápidamente a las solicitudes del cliente en el formulario de consultas SQL. Estas consultas se pueden generar directamente en SQL Server, o por medio de una interfaz de usuario independiente desarrollada en una variedad de lenguajes de programación. SQL Server está diseñado para manejar bases de datos con millones de registros.
- **Oracle:** Es otra base de datos escalable a nivel empresarial. La base de datos de SQL soporta bases de datos corporativos distribuidos, que permiten al usuario acceder a los datos de forma local o desde bases de datos remotas en una transacción transparente. Las bases de datos distribuidas ayudan a superar las limitaciones físicas de un entorno informático físico. El tamaño máximo de base de datos para una Oracle es de 8 millones de terabytes, lo que requiere un almacenamiento físico más allá de la capacidad de la mayoría de las instalaciones individuales.

Las etapas del ciclo de vida de una aplicación de bases de datos son las siguientes:

1. Planificación del proyecto.
2. Definición del sistema.
3. Recolección y análisis de los requisitos.
4. Diseño de la base de datos.
5. Selección del DBMS.
6. Diseño de la aplicación.
7. Prototipado.
8. Implementación.
9. Conversión y carga de datos.
10. Prueba.
11. Mantenimiento.

3.4 Modelo Relacional

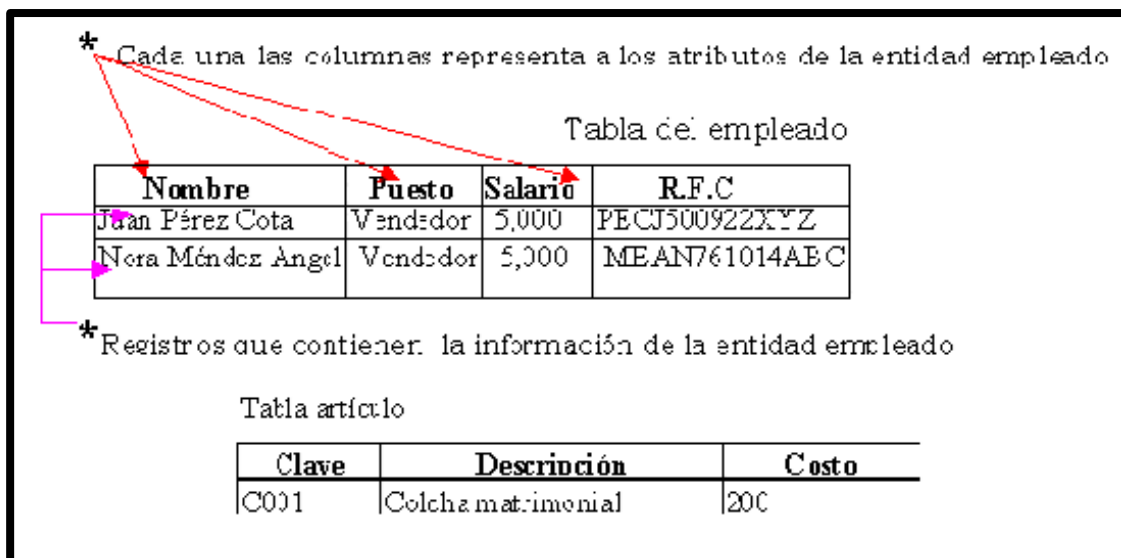
Principios básicos de diseño

- Todos los datos se representan en tablas.
- Incluso los resultados de cualquier consulta son otra tabla.
- Las tablas están compuestas por filas y columnas.
- Las filas y las columnas, en principio, carecen de orden.
- Las filas sólo se ordenan si se le indica a la base de datos que lo haga, mediante el correspondiente comando.
- De no ser así, el orden será arbitrario, y puede cambiar en caso de tratarse de una base datos dinámica.

- El orden de las columnas lo determina cada consulta.
- Cada tabla tiene una clave primaria, un identificador único, compuesto por una o más columnas.
- La mayoría de las claves primarias están formadas por una única columna.
- Para establecer una relación entre dos tablas es necesario incluir, en forma de columna, en una de ellas la clave primaria de la otra. A esta columna se le llama clave secundaria.
- Estos dos conceptos --clave primaria y secundaria-- son los más importantes en el diseño de bases de datos.

Por Ejemplo:

Tabla del empleado



Cualidades de un buen diseño de base de datos

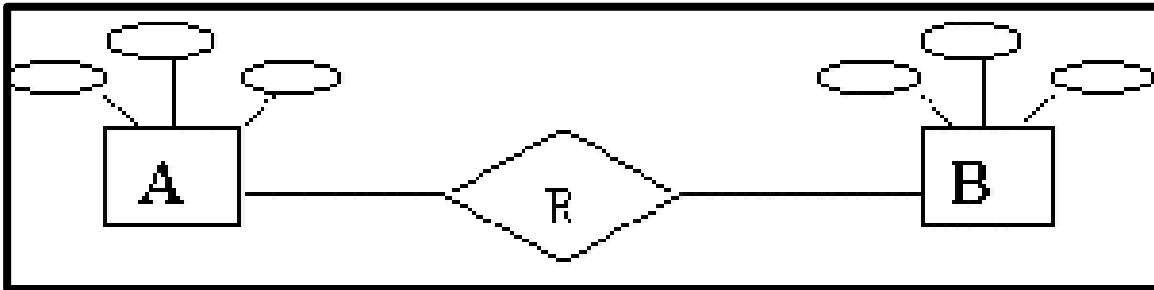
- Reflejar la estructura del problema en el mundo real.
- Ser capaz de representar todos los datos esperados, incluso con el paso del tiempo.
- Evitar el almacenamiento de información redundante.
- Proporcionar un acceso eficaz a los datos.
- Mantener la integridad de los datos a lo largo del tiempo.
- Ser claro, coherente y de fácil comprensión.

Definir modelo Entidad/Interrelación (E/R)

- El modelo Entidad/Interrelación (E/R): un método de diseño de bases de datos.
- Muestra de una versión simplificada.
- Representa los datos mediante una serie de entidades que disponen de atributos.
- Una entidad es una clase de objetos o conceptos claramente identificable.
- Las entidades establecen interrelaciones con otras entidades.
- El resultado de este proceso es una base de datos normalizada que facilita el acceso a los datos y evita su duplicado.

Proceso de diseño en el modelo E-R

- Identificar las entidades que debe presentar la base de datos.
- Determinar las cardinalidades de las interrelaciones establecidas entre las distintas entidades y clasificar estas interrelaciones entre los siguientes tipos:
 - Uno a uno.
 - Uno a muchos.
 - Muchos a muchos (p.ej., la venta de parcelas: una misma parcela la pueden vender varios propietarios y cada propietario puede vender varias parcelas).
 - Dibujar el diagrama Entidad/Interrelación.



- Determinar los atributos de cada entidad.
- Definir la clave primaria (única) de cada entidad

Paso del modelo E-R al diseño de la base de datos

- Las entidades entre las que hay una interrelación uno a uno se deben fusionar en una sola entidad.
- Una vez hecho esto, cada una de las entidades que quedan se convierte en una tabla con una clave primaria y una serie de atributos, de los cuales algunos pueden ser claves secundarias.
- Las interrelaciones uno a muchos se transforman en atributo y clave secundaria de la tabla que representa a la entidad situada del lado de la interrelación correspondiente a muchos.
- Las interrelaciones muchos a muchos entre dos entidades pasan a ser una tercera tabla con claves secundarias procedentes de ambas entidades. Estas claves secundarias deberán formar parte de la clave primaria de la tabla en la que se convierte la interrelación, cuando corresponda.
- Hay una serie de herramientas disponibles en el mercado que pueden automatizar el proceso de conversión de un modelo E/R en un esquema de base de datos.

Por ejemplo:

