

Bases de Datos II

Unidad 1 – Administración de BD

Objetivos

- Conocer las tareas que realiza un Administrador de BD.
 - Conocer lo que realiza un Software de Administración de BD.
 - Primeros conceptos acerca de la administración de Bases de Datos
 - Importancia del Administrador de BD (DBA).
 - Herramientas de un Sistema de Administración de BD.
 - Elección de un Sistema de Administración de BD.
 - Evolución de los Sistema de Administración de BD.
-

Conceptos Básicos

- Base de Datos: es una colección de archivos, datos, información; ordenada, organizada, y relacionada, con la finalidad de permitir el manejo de la información para su procesamiento. Cada uno de los archivos representan una colección de registros y cada registro está compuesto de una colección de campos. Cada uno de los campos de cada registro permite llevar información de alguna característica o atributo de alguna entidad del mundo real.
 - DBMS: es un conjunto de programas que se encargan de manejar la creación y todos los accesos a las bases de datos. Se compone de un Lenguaje de Definición de Datos (DDL: Data Definition Language), de un Lenguaje de Manipulación de Datos (DML: Data Manipulation Language), y de un Lenguaje de Consulta (SQL: Structured Query Language).
 - DBA: es el responsable del control general del sistema a nivel técnico.
-

Funciones del DBA

- Definir el esquema conceptual.
 - Definir el esquema interno.
 - Establecer un enlace con los usuarios.
 - Definir las restricciones de seguridad e integridad.
 - Definir las políticas de backup y recuperación.
 - Supervisar la performance y responder a los requerimientos cambiantes.
-

Funciones del DBMS

- Definición de datos.
 - Manipulación de datos.
 - Optimización y ejecución.
 - Seguridad e Integridad de datos.
 - Recuperación de datos y concurrencia.
 - Diccionario de datos.
 - Concurrencia.
-

DBMS

El DBMS sirve como interfase entre la base de datos física y las peticiones del usuario. El DBMS interpreta las peticiones de entrada/salida del usuario y las manda al sistema operativo para la transferencia de datos entre la unidad de memoria secundaria y la memoria principal. En sí, un sistema manejador de base de datos es el corazón de la base de datos ya que se encarga del control total de los posibles aspectos que la puedan afectar.



Implementación del Esquema Conceptual

- Describe la estructura lógica global de la base de datos mediante un modelo abstracto de datos comprensible por el SGBD. Se definen la descripción de atributos, de entidades, las conexiones y las restricciones de integridad asociadas a la semántica (significado). Podemos decir que describe que datos son almacenados realmente en la base de datos y las relaciones que existen entre los mismos, describe la base de datos completa en términos de su estructura de diseño. El nivel conceptual de abstracción lo usan los administradores de bases de datos, quienes deben decidir qué información se va a guardar en la base de datos.
- El esquema conceptual representa la visión organizacional de la base de datos que se obtiene al integrar los requerimientos de todos los usuarios en una empresa; y es totalmente independiente de las estructuras físicas de almacenamiento y de la representación final de los datos que los usuarios manejan. La implantación de este esquema es responsabilidad del DBA.

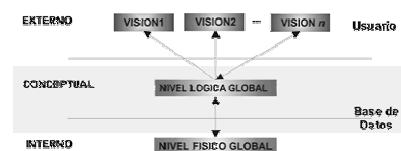
Implementación del Esquema Conceptual

Definición de los datos: Se describen el tipo de datos y la longitud de campo todos los elementos direccionables y precisión de la información que será guardada en la base de datos. Los elementos por definir incluyen artículos elementales (atributos), totales de datos y registros conceptuales (entidades).

Relaciones entre datos: Se definen las relaciones entre datos para enlazar tipos de registros relacionados para el procesamiento de archivos múltiples para obtener información compuesta y procesos complejos.

En el nivel conceptual la base de datos aparece como una colección de registros lógicos, sin descriptores de almacenamiento. En realidad los archivos conceptuales no existen físicamente. La transformación de registros conceptuales a registros físicos para el almacenamiento se lleva a cabo por el sistema y es transparente al usuario.

Diagrama del esquema conceptual



Implementación del Esquema Interno

- Es el nivel más bajo de abstracción, describe que datos son almacenados realmente en la base de datos y las relaciones que existen entre los mismos, describe la base de datos completa en términos de su estructura de diseño. El diseño físico es el proceso de producir la descripción de la implementación de la base de datos en memoria secundaria: estructuras de almacenamiento y métodos de acceso que garanticen un acceso eficiente a los datos. Entre el diseño físico y el diseño lógico hay una realimentación, ya que alguna de las decisiones que se tomen durante el diseño físico para mejorar las prestaciones, pueden afectar a la estructura del esquema lógico.

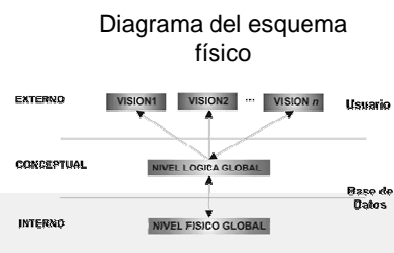
Implementación del esquema físico

El propósito del diseño físico es describir cómo se va a implementar físicamente el esquema lógico obtenido en la fase anterior.

- Obtener un conjunto de relaciones (tablas) y las restricciones que se deben cumplir sobre ellas.
- Determinar las estructuras de almacenamiento y los métodos de acceso que se van a utilizar para conseguir unas prestaciones óptimas.
- Diseñar el modelo de seguridad del sistema.

En el nivel físico se debe especificar:

- Estrategias de Almacenamiento: Asignación de espacio de almacenamiento para el conjunto de datos.
- Estrategias de Emplazamiento de los Datos: Para optimizar los recursos a la hora de exportar la base de datos (tiempo de respuesta, disco, memoria, etc.).
- Caminos de Acceso: Se incluye la especificación de claves como la de índices y punteros.



Implementación de los esquemas externos

- Es el nivel más alto de abstracción, es lo que el usuario final puede visualizar del sistema terminado, describe sólo una parte de la base de datos al usuario acreditado para verla. El sistema puede proporcionar muchas visiones para la misma base de datos.
- Las **Vistas**: Son una especie de tablas virtuales; es decir no existen físicamente sino que forman mediante la selección y/o filtrado de los componentes de otras tablas, una vista puede ser definida en base a una lista previa. Esto significa que pueden crearse dependencia entre las vistas. Cuando una vista es definida en base a otra, se dice que es dependiente de esta por lo tanto, se suprimirá automáticamente la vista dependiente si se suprime la vista original.

Implementación de los esquemas externos

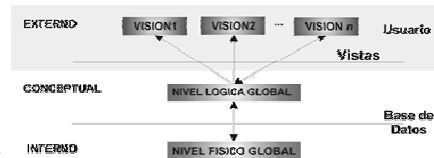
Objetivos y Ventajas de las Vistas.

Facilitar al usuario la percepción que este tiene de la base de datos así como el trabajo que van a desarrollar sobre ésta.

Ventajas que se obtienen al utilizar vistas son:

- **Perspectivas Directas:** Proporcionarse diversos modelos de información basados en los mismos datos, enfocándolos hacia distintos usuarios con necesidades específicas. El mostrar la información desde distintos ángulos nos ayuda a crear ambientes de trabajo y operación acordes a los objetivos de la empresa.
- **Transparencias en las Modificaciones:** El usuario final no se verá afectado por el diseño o alteraciones que se realicen en el esquema conceptual de la base de datos.
- **Seguridad:** Las vistas proporcionan de manera natural un medio para ocultar y proteger datos, dado que solo se presenta al usuario una selección de los atributos existentes.

Esquema de Nivel de Vistas



Diccionario de Datos

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

Diccionario de Datos

- Es un catálogo, un depósito, de los elementos en un sistema. Contiene las características lógicas de los sitios donde se almacenan los datos del sistema, incluyendo nombre, descripción, alias, contenido y organización. Identifica los procesos donde se emplean los datos y los sitios donde se necesita el acceso inmediato a la información, se desarrolla durante el análisis de flujo de datos y auxilia a los analistas que participan en la determinación de los requerimientos del sistema, su contenido también se emplea durante el diseño.
- En un diccionario de datos se encuentra la lista de todos los elementos que forman parte del flujo de datos en todo el sistema. Los elementos más importantes son flujos de datos, almacenes de datos y procesos. El diccionario guarda los detalles y descripciones de todos estos elementos.
- Si los analistas desean conocer cuántos caracteres abarca un determinado dato o qué otros nombres recibe en distintas partes del sistema, ~~o dónde se utiliza, encontrarán las respuestas en un~~ diccionario de datos desarrollado en forma apropiada.

Ventajas del Uso de un DD

1. Para manejar los detalles en sistemas muy grandes, ya que tienen enormes cantidades de datos, aun en los sistemas más chicos hay gran cantidad de datos. Los sistemas al sufrir cambios continuos, es muy difícil manejar todos los detalles.
2. Para asignarle un solo significado a cada uno de los elementos y actividades del sistema. Los diccionarios de datos proporcionan asistencia para asegurar significados comunes para los elementos y actividades del sistema y registrando detalles adicionales relacionados con el flujo de datos en el sistema, de tal manera que todo pueda localizarse con rapidez.
3. Para documentar las características del sistema, incluyendo partes o componentes así como los aspectos que los distinguen. También es necesario saber bajo que circunstancias se lleva a cabo cada proceso y con que frecuencia ocurren. Produciendo una comprensión mas completa.
4. Para facilitar el análisis de los detalles con la finalidad de evaluar las características y determinar donde efectuar cambios en el sistema. Determina si son necesarias nuevas características o si están en orden los cambios de cualquier tipo. Se abordan las características:
 1. **Naturaleza de las transacciones:** las actividades de la empresa que se llevan a cabo mientras se emplea el sistema.
 2. **Preguntas:** solicitudes para la recuperación o procesamiento de información para generar una respuesta específica.
 3. **Archivos y bases de datos:** detalles de las transacciones y registros maestros que son de interés para la organización.
 4. **Capacidad del sistema:** Habilidad del sistema para aceptar, procesar y almacenar transacciones y datos.
5. Localizar errores y omisiones en el sistema, detectar dificultades, y las presentan en un informe.

Conceptos Básicos

Un sistema de información es un conjunto de herramientas a través de las que se generan y gestionan datos, diferentes actores interactúan con él a través de distintos perfiles. Ese conjunto de datos, que forman el núcleo del sistema de información, deben ser almacenados en un repositorio confiable capaz de resguardarlos asegurando la integridad y la seguridad de los mismos. Los sistemas de información modernos utilizan Bases de Datos para tal fin.

Conceptos Básicos

Base de Datos es un software capaz de almacenar en forma confiable y segura la información, manteniéndola disponible para que pueda ser recuperada cuando los usuarios del sistema lo soliciten.

En la actualidad, el valor de la información hizo que los sistemas de gestión de los datos ocupen un rol fundamental en la vida de toda organización. Por este motivo surge la necesidad de contar con personal adecuado que se encargue de llevar adelante la administración de los sistemas y en particular de sus bases de datos.

Para asegurar el correcto funcionamiento de la base y optimizar el acceso a los datos, es imprescindible la figura de un administrador de bases de datos o DBA (por sus siglas en inglés Data Base Administrator).

Funciones del DBA

- En principio el DBA debe conocer a la perfección el lenguaje que utiliza el sistema para comunicarse con la base de datos, cuando un usuario necesita agregar, modificar o extraer información en el sistema, denominado SQL (por su sigla en inglés Structured Query Language).
 - Otra de sus tareas será definir, de acuerdo a las características de cada herramienta y de la institución en la que se implementará, las políticas de seguridad y asegurar la performance de la base. En cuanto a la performance, es el encargado de realizar todas las tareas de mantenimiento, tanto de hardware como de software, para que la base de datos responda de manera eficiente a las diferentes consultas, esto incluye actualizaciones, configuraciones, etc.
-

Funciones del DBA

- Respecto a la seguridad, uno de los aspectos más importantes es definir las políticas de acceso a la información, es decir que el DBA debe tener la capacidad de configurar los accesos para que cada usuario vea sólo lo que su rol le permite. Otro aspecto a tener en cuenta es definir qué tipo de políticas de resguardos o backups se utilizará, y cuáles serán los pasos a seguir ante un fallo de hardware o de software, para asegurar que no se pierda la información almacenada.
- Si bien el trabajo más fuerte del DBA se sitúa al momento de la puesta en marcha de un proyecto, cabe destacar que su rol es continuo, ya que no se centra en una parte específica del proceso de evolución de los sistemas, es decir que se encuentra presente desde el desarrollo (control del modelo conceptual, físico y esquemas externos) hasta la implementación y durante su posterior uso. Una vez puesto en marcha el sistema, el DBA será el encargado de monitorear constantemente el funcionamiento de las bases realizando tareas preventivas y correctivas en los procesos.
- El rol del DBA excede el alcance de un sistema, por lo que deberá tener una visión integral de todos los módulos que integran un único sistema de información.

Herramientas de un DBMS

- Afinación.
 - Medición del desempeño.
 - Reorganización Física y Lógica.
-

Afinación (Tunning)

- Ajustes y cambios en 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).
- El uso de la base de datos evoluciona continuamente, a medida que más personas se van familiarizando con ella y se van creando más programas de aplicación. Los ajustes en la organización del almacén para el óptimo desempeño se convierten en un **proceso continuo**.
- El responsable de realizar la afinación de la base de datos es el DBA o su grupo, y es importante que este tenga libertad para introducir los cambios que sean necesarios.

Requisitos para una correcta afinación:

- **Independencia física de los datos.** Es la capacidad de modificar el esquema interno sin alterar el esquema conceptual, ni los programas de aplicación.
- **Medios:** Para supervisar automáticamente el uso de la base de datos con el fin de que puedan hacerse los ajustes necesarios.

Los manejadores actuales de bases de datos ya incorporan medios para la afinación automática.

Medición del desempeño

- Es responsabilidad del DBA organizar el sistema de modo que se obtenga el desempeño que sea "mejor para la empresa", y realizar los ajustes apropiados cuando cambien los requerimientos. Es necesario reorganizar la base de datos (descargarla y volverla a cargar) en forma periódica con el fin de garantizar que los niveles de desempeño sigan siendo aceptables.
- Los datos obtenidos del desempeño se comparan con aquellos datos esperados. Esta razón proporciona los factores de multiprogramación, M_n , para cada uno de los procesos básicos h que son parte de los cálculos de la base de datos. En la práctica estos factores pueden variar de 1 a 0.1.
- Para estas pruebas no se requiere la existencia de la base de datos completa y puede ser que ni siquiera exista el estudio piloto, que consiste en la ejecución de una secuencia adecuada de operaciones aleatorias o secuenciales de lectura o escritura, junto con una cantidad equivalente del uso del cpu. Es posible escribir y ejecutar un programa que imite la operación propuesta, en el sistema real que se va a utilizar, ~~con el costo del esfuerzo de unos cuantos días.~~

Reorganización Física y Lógica

- La reorganización consiste en leer el archivo en forma en que se utilizaría al realizar el procesamiento serial y escribir los registros nuevos y viejos en el archivo nuevo, dejando fuera todos los registros marcados como eliminados lógicamente; y se crearán nuevos índices con base a los nuevos valores.
- La frecuencia de reorganización depende de la actividad de inserción dentro del archivo; y se debe realizar antes de que el archivo esté realmente lleno para evitar problemas en tiempos de mucha actividad. Las reorganizaciones físicas son necesarias para mejorar el rendimiento, añadir una nueva estructura de acceso, agilizar las operaciones de obtención y actualización, disminuir los tiempos de respuesta, minimizar el espacio de almacenamiento y optimizar el consumo de recursos. Las reorganizaciones lógicas pueden modificar el esquema conceptual, pero no alterar el esquema externo ni los programas de aplicación; puede ser un orden de visualización en las vistas.

Respaldo y Recuperación

- El DBA debe definir y poner en práctica un plan (política) de recuperación adecuado que incluya una descarga o vaciado periódico de la base de datos en un medio de almacenamiento de respaldo y procedimientos para volver a cargar la base de datos a partir del vaciado más reciente. El respaldo y recuperación consiste en contar con un mecanismo que permiten la fácil recuperación de los datos en caso de ocurrir fallos en el sistema de la base de datos.
- El objetivo del concepto de recuperación es el de proteger la base de datos contra fallas lógicas y físicas que destruyan los datos en todo o en parte independientemente de la naturaleza de las fallas estas pueden afectar los aspectos de almacenamiento de la base de datos como son:
 - Fallas que provocan la pérdida de la memoria volátil.
 - Fallas que provocan la pérdida del contenido de la memoria secundaria.
- En un sistema de base de datos, recuperación significa, restaurar la base de datos a un estado que se sabe que es correcto, después de una falla que provoca que se considere que el estado actual es incorrecto. Podemos tener la seguridad de que la base de datos es recuperable, si aseguramos que cualquier parte de la ~~información que contiene, puede ser reconstruida, a partir de otra información que se encuentra almacenada redundantemente en algún lugar del sistema.~~

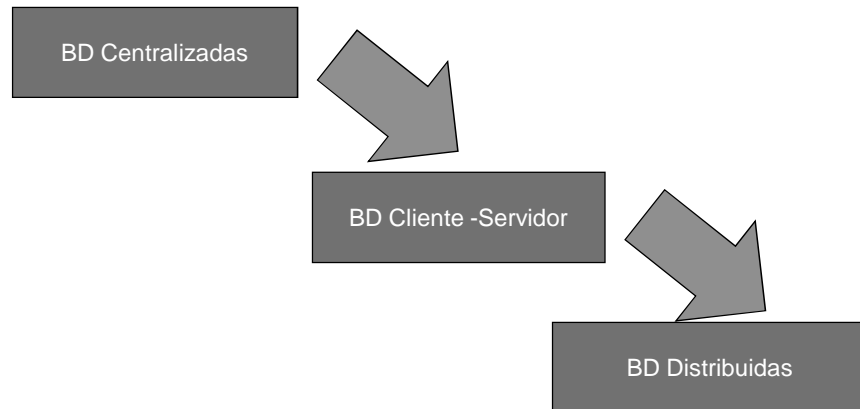
Pasos para recuperar la información

- **Detección del error.-** El proceso de recuperación se inicia al detectar la existencia de un error. Es posible distinguir una variedad de puntos de entrada en el proceso de recuperación. Se considerarán fallas de sistemas detectadas por falta de acción del sistema o por verificaciones irrecuperables de redundancia y salida incorrecta observada por un usuario.
- **Determinación de la fuente del error.-** para decidir cual es la mejor acción correctora es necesario determinar la extensión del daño. Desde luego este esfuerzo es muy relacionado con la determinación del tiempo y la causa del error. Después de una caída o cuando el procesamiento sea interrumpido debido a una señal de error, es necesario determinar tantas aquellas áreas del archivo de datos que sean sospechosas como cuál fue la transacción que no se concluyó.
- **Ubicación de errores secundarios.-** cuando se ha detectado un error que provocó una modificación inadecuada a un archivo un rastreo a través de las listas de actividad encontrara aquellas transacciones que emplearon el bloque correcto. Entonces es posible volver a introducir automáticamente el bloque correcto de las transacciones afectadas y producir resultados correctos. Si se actualizaron bloques mediante transacciones que leyeron bloques incorrectos antes de existir es necesario restaurar a un más el archivo.
- **Aplicación de correcciones.** Si la extensión del daño es limitada, puede utilizarse un ~~proceso de volver a aplicar. Las porciones dañadas del archivo se restauran aplicando~~ primero aquellas imágenes anteriores a los bloques en error reemplazando después de las transacciones incompletas.

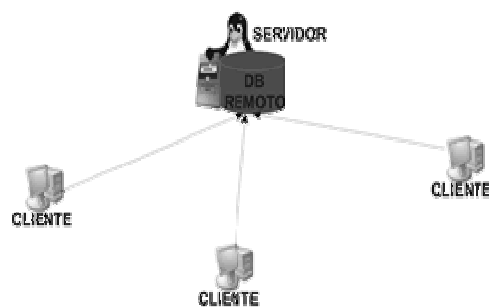
Evolución de los DBMS

- La evolución de los sistemas de información y el crecimiento no planeado de la información dentro de las organizaciones, ha traído dispersión de los datos en sitios local o geográficamente dispersos. La necesidad de integrar y compartir dicha información implica el nacimiento de una nueva tecnología capaz de conformar de manera consistente la información de las organizaciones. Una de las tecnologías que trabaja en el problema de integración de información, es la de bases de datos distribuidas (BDD).
-

Evolución de los DBMS



1º Etapa: BD Centralizada



Los equipos están conectados a un servidor centralizado, donde esta almacenada la base de datos. La interacción con el usuario como la aplicación residen en el cliente, siendo el servidor el depositario de los datos. En un sistema centralizado, la caída o desconexión del servidor, provoca la caída total del sistema.

2º Etapa: Arquitectura Cliente-Servidor



Es la integración distribuida en un sistema de red, con los recursos, medios y aplicaciones, que definidos modularmente en los servidores, administran, ejecutan y atienden a las solicitudes de los clientes; todos interrelacionados física y lógicamente.

En un esquema cliente-servidor, se denomina *cliente* a la máquina que solicita un determinado servicio, y *servidor* a aquella que proporciona tal servicio.

Arquitectura Cliente-Servidor Características

- Varias computadoras están conectadas por medio de una red
 - El sistema es abierto esto implica que las computadoras pueden ser de diferente tipo y/o marca corriendo diferentes S.O.
 - Por lo general las computadoras que procesan los programas de aplicación se llaman "clientes"
 - La computadora que soporta el DBMS y el servicio de los datos se llama "servidor"
 - La comunicación entre el cliente y el servidor se hace, por lo general, por medio Remote Procedure Calls (RPC) o servicios de SQL
 - Existen dos variantes principales de esta arquitectura:
 - Cliente-Servidor de 2 capas (bandas) (Two Tier architecture)
 - Cliente-Servidor de 3 capas (bandas) (Three Tier architecture)
-

Cliente-Servidor de dos capas

- Primera capa : lógica de las aplicaciones
- Segunda capa: administración de recursos de datos (DBMS)

Propósito y Origen

- Se desarrolló en lo 80's
 - La intención mejorar la "usabilidad" por medio del soporte de interfaces gráficas (GUI) "amigables" para el usuario
 - Mejorar la "escalabilidad" permitiendo trabajar con cierta comodidad hasta 100 usuarios
 - Mejorar la flexibilidad permitiendo que los datos puedan ser compartidos usualmente dentro de ambientes homogéneos.
-

Cliente-Servidor de dos capas

Detalles técnicos

- Interfase de Usuarios del Sistema: sesión, entrada/salida de datos, diálogo.
 - Administración de los Procesos: desarrollo y verificación de procesos, servicios y recursos de los procesos
 - Administración de la Base de Datos: servicios de archivos y datos. Los servicios GUI (1) se asignan exclusivamente al cliente, los del DBMS al servidor (3) y la administración del procesamiento (2) se distribuye en ambas capas.
 - Es muy común que en un cliente servidor de dos capas, las reglas del negocio se efectúen como procedimientos y triggers almacenados en el servidor, y el procesamiento de la lógica de la aplicación en el ambiente del cliente.
 - USO: Se emplean extensivamente en procesamiento de información cuyo procesamiento no es crítico en tiempo y donde la administración y operación del sistema no es complejo, y en donde la carga de transacciones no es alta
-

Cliente-Servidor de dos capas

Costos y limitaciones

- Escalabilidad: se pueden manejar bien hasta aproximadamente 100 usuarios. Si las reglas del negocio + procedimientos + funciones se alojan en el servidor entonces se reduce el número de usuarios que se pueden agregar posteriormente.
 - Interoperabilidad: cuando se emplean procedimientos almacenados para implementar reglas de negocio complejas la posibilidad de operar con DBMS distintos se ve limitada (procedimiento en lenguajes propietarios con poca capacidad de "debugging").
 - Administración y configuración: cuando las aplicaciones residen en el cliente las actualizaciones se deben instalar y testear en cada cliente. La típica falta de uniformidad en la configuración de cada cliente y la falta de control sobre los cambios de configuración subsecuentes incrementa la carga de trabajo.
-

Cliente-Servidor de dos capas

Ventajas

- División del procesamiento entre el sistema cliente y el servidor de bases de datos
- Sistema abierto los clientes y los servidores no están atados por un modelo/marca/SO
- Independencia de las aplicaciones, los usuarios pueden emplear el software que les resulte mas familiar para acceder al DBMS

Desventajas

- Mantenimiento de los clientes (aplicaciones y hardware)
 - Costo en aprendizaje y en personal (heterogeneidad de los componentes)
 - Configuración de los distintos componentes del sistema y hacer que funcionen coordinadamente es complejo
-

Cliente-Servidor de tres capas

Propósito y Origen

- Se desarrolló en el principio de los 90's
 - Intentaba cubrir las falencias del modelo de dos capas
 - Mejorar la "escalabilidad" permitiendo trabajar con cierta comodidad hasta miles de usuarios
 - Mejorar la flexibilidad permitiendo que los datos puedan ser compartidos en ambientes heterogéneos.
 - Mejorar el mantenimiento de los clientes, ubicando gran parte del procesamiento en la capa del medio haciendo los clientes más "flacos"
-

Cliente-Servidor de tres capas

Detalles técnicos

- La capa del medio que se ubica entre el cliente y el servidor de datos es el lugar donde se ubican la administración de los procesos y las reglas de negocio (procedimientos y triggers del modelo de dos capas).
 - La capa del medio administra las funciones de administración de la base de datos (servicios de datos y archivos) sin usar lenguajes propios del DBMS.
 - La capa del medio también se ocupa de la administración de los procesos, desarrollo y verificación de procesos, servicios y recursos de los procesos
 - El servidor de la capa del medio, también referido como el Servidor de Aplicaciones ("Application Server") es el que centraliza la lógica del proceso. La centralización hace que la lógica de los procesos y la administración de los cambios sea más fácil
 - La capa del medio hace un control de las transacciones y pone en "cola" las mismas para asegurar que las transacciones lleguen a su fin de manera segura
-

Cliente-Servidor de tres capas

- En algunos casos la capa del medio se divide en dos ó mas unidades con diferentes funciones y la arquitectura aquí se dice que es "multicapa". Es el caso de aplicaciones de Internet, cuyos clientes son "flacos" escritos en HTML y cuentan con un servidor de aplicaciones escritos en C++ o Java.
 - El "link" entre las dos aplicaciones la hace un "Web Server" que recibe los requerimientos de los usuarios y genera documentos HTML en conjunto con los servicios que le provee el resto de los servidores. Esta es la arquitectura de una aplicación Intranet/Extranet.
 - USO: Se emplean en ambientes cliente/servidor distribuidos donde se necesita compartir recursos tales como DBMS heterogéneos y reglas de negocios.
-

Cliente-Servidor de tres capas

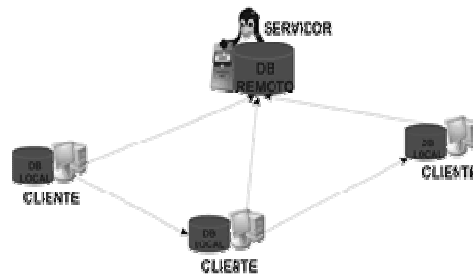
- La arquitectura de tres capas facilita el desarrollo porque cada capa puede ser construida y ejecutada sobre una plataforma diferente, haciendo la organización e implementación más fácil.
 - Permite, además que las capas se puedan desarrollar en diferentes lenguajes: Lenguaje GUI, clientes Internet "flacos" (HTML) para la capa del cliente
 - C, C++, Java, VisualBasic, SmallTalk, ADA para la capa del medio
 - SQL para la capa de la base de datos
 - El modelo de tres capas facilita la migración de las aplicaciones, ya que el sistema nuevo podría ir generándose y ejecutándose en paralelo con el nuevo hasta que el sistema se haya migrado completamente
-

Cliente-Servidor de tres capas

Costos y limitaciones

- La construcción de una arquitectura de tres capas es un trabajo complejo
 - Las herramientas de programación de la capa del medio no proveen aún de todas las funcionalidades deseables de un ambiente de computación distribuido
 - Un problema que se presenta en el desarrollo de las aplicaciones es la separación de la lógica del cliente, la separación entre la lógica del proceso y la lógica de los datos no es siempre clara.
 - Los criterios para juzgar la ubicación de una función en una capa determinada son los siguientes:
 - fácil de desarrollar y testear
 - fácil de administrar
 - escala del servidor
 - performance del procesamiento y de la red
-

3º Etapa:BD Distribuidas



Una base de datos esta dividida en fragmentos, almacenados en diferentes sitios de la red. Todas las operaciones realizadas en los sitios se ven reflejadas en la base de datos que esta almacenada en el servidor.

El cliente debe conocer la topología de la red, así como la disposición y ubicación de los datos. Se delega parte de la gestión de la base de datos a los clientes.

BD Distribuidas - Definición

Un Sistema de Bases de Datos Distribuidas es un conjunto de sitios autónomos de procesamiento de datos que se encuentran a una distancia determinada unos de otros conectados por una red de comunicaciones de datos y cooperando entre sí.

- Un DDBMS debe proveer acceso a los datos distribuidos
 - Los datos físicos están distribuidos en computadoras diferentes, pero el sistema presenta una sola imagen al usuario
 - Técnicamente es muy difícil proveer una imagen simple de la Base de Datos y pocos proveedores presentan una solución completa
 - El DDBMS debe proveer el acceso a los datos como si ellos estuvieran en la computadora local
 - No se comparte memoria principal
-

BD Distribuidas - Ventajas

Mayor paralelismo

- Al estar los datos distribuidos en varias localidades una consulta puede ser subdividida en varias subconsultas que se ejecuten en paralelo

Mayor independencia

- Cada localidad puede administrar sus propios datos de manera local y, por lo tanto, definir sus estrategias de acceso, seguridad, integridad, etc.

Mayor flexibilidad

- La introducción de cambios es más sencilla en un sistema distribuido. Es más fácil agregar nuevas localidades si es necesario, que mover o cambiar una computadora centralizada.

Mayor disponibilidad

- Si se produce una falla en una localidad las demás localidades puedan seguir trabajando. Además, si los datos están duplicados en otras localidades, es posible encontrar el dato que se necesita en otro lado.

BD Distribuidas - Desventajas

Mayor costo/complejidad

- El desarrollo del software en un sistema distribuido es mas difícil de desarrollar y estructurar, por lo tanto es más complejo y su costo es mayor.
- Además hay que sumarle el costo de comunicación y de almacenamiento adicional de las copias

Mayor dificultad en el control

- Al tener las localidades que trabajan en paralelo, y copias de la base de datos distribuida en diferentes localidades es mas difícil de garantizar que los algoritmos sean correctos

Mayor riesgo de seguridad

- Al estar el sistema distribuido, es mas vulnerable a que existan sabotajes, pérdidas y fugas de información.
-

Objetivos

- Conocer las tareas que realiza un Administrador de BD.
 - Conocer lo que realiza un Software de Administración de BD.
 - Primeros conceptos acerca de la administración de Bases de Datos
 - Importancia del Administrador de BD (DBA).
 - Herramientas de un Sistema de Administración de BD.
 - Elección de un Sistema de Administración de BD.
 - Evolución de los Sistema de Administración de BD.
-



¿Dudas, Consultas?

Base de Datos II