



Preparador Informática

TEMA 55. INFORMÁTICA.

**DISEÑO FÍSICO DE DATOS Y
FUNCIONES. CRITERIOS DE DISEÑO.
DOCUMENTACIÓN**

TEMA 55. DISEÑO FÍSICO DE DATOS Y FUNCIONES. CRITERIOS DE DISEÑO. DOCUMENTACIÓN

1. INTRODUCCIÓN

2. DISEÑO FÍSICO DE DATOS. CRITERIOS DE DISEÑO

2.1. OBJETIVOS Y CRITERIOS DE DISEÑO

2.2. FACTORES QUE INFLUYEN EN EL DISEÑO FÍSICO

2.3. FASES DEL DISEÑO FÍSICO

3. DISEÑO FÍSICO DE FUNCIONES. CRITERIOS DE DISEÑO

3.1. CRITERIOS DE DISEÑO

3.2. FASES DEL DISEÑO FÍSICO

4. DOCUMENTACIÓN

5. CONCLUSIÓN

6. BIBLIOGRAFÍA

Preparador Informática



1. INTRODUCCIÓN

Habitualmente para llevar a cabo la construcción de un sistema informático complejo se utiliza una metodología de diseño en varias fases o modelos: modelo conceptual, modelo lógico y modelo físico.

El diseño conceptual permite plasmar una descripción de alto nivel del sistema, independientemente de las tecnologías que se vayan a llevar a cabo para implementarlo. Su objetivo es describir el contenido del sistema de manera abstracta y cercana al usuario.

El diseño lógico parte del resultado del diseño conceptual y da como resultado un conjunto de componentes de negocio y sus servicios. Se centra en la identificación de los elementos que formarán parte del sistema y cómo se relacionan entre ellos.

En el diseño físico se da forma al diseño lógico para adaptarlo a una tecnología concreta y se completa con aspectos de implementación física de esa tecnología. La fase de diseño físico constituye la fase previa a la implementación física de la aplicación en un sistema computacional concreto. Se recoge pues todas las orientaciones y trabajos previos de las fases de análisis y diseño lógico, suponiendo la confirmación o refutación de su validez y dividiéndose también en diseño físico funcional y diseño físico de datos.

Precisamente esto último es lo que se expone en este tema, el cómo realizar el diseño físico de funciones y el diseño físico de datos especificando los criterios de diseño para cada uno de ellos, así como la documentación creada en esta fase.

2. DISEÑO FÍSICO DE DATOS. CRITERIOS DE DISEÑO

Cuando comenzamos a definir el diseño de una base datos, debemos tener en cuenta que este se encuentra dividido en 3 fases: diseño conceptual, diseño lógico y finalmente una referente a su diseño físico. Cada fase es independiente y se encuentran relacionadas al mismo tiempo, por lo cual pueden ser creadas por separado, pero necesitan tener ciertos estándares que les permitan pasar de una fase a otra.

En este tema nos concierne el diseño físico de datos que es el proceso por el cual a partir de un esquema lógico se obtiene el esquema físico de la base de datos. Mientras que en el diseño lógico se especifica qué se guarda, en el diseño físico se especifica cómo se guarda. Esta descripción incluye las estructuras de almacenamiento y los métodos de acceso que se utilizarán para conseguir un acceso eficiente a los datos.

El esquema físico es una descripción del modo de implantar una base de datos en un SGBD por lo que es muy dependiente no sólo del modelo lógico de base de datos utilizado (relacional, orientado a objetos...) sino del SGBD concreto que se vaya a utilizar.

Además, es necesario tomar decisiones que optimicen el rendimiento de la BD y proporcionen la máxima seguridad, a costa de modificar el esquema lógico si fuera necesario. Una vez completado el proceso obtendremos un esquema físico, que se expresa mediante el Lenguaje de Definición de Datos (DDL) propio del SGBD a utilizar.

2.1. OBJETIVOS Y CRITERIOS DE DISEÑO

El objetivo general del diseño físico es determinar estructuras de almacenamiento y estructuras de acceso para que las aplicaciones que accedan a la BD obtengan un buen rendimiento. Todo ello se puede concretar en distintos objetivos:

- Minimizar el tiempo de respuesta: (El tiempo de respuesta es el tiempo entre la introducción de una transacción T de BD y la obtención de respuesta). Depende de (entre otras cosas):
 - Tiempo de acceso a la BD para obtener datos que la transacción necesita: esto sí está bajo el control del SGBD.
 - Carga del sistema, planificación de tareas del SO ó retrasos de comunicación: esto está fuera del control del SGBD.

- Maximizar la productividad de las transacciones:
 - o Número medio de transacciones que SGBD puede procesar por minuto.
 - o Es un parámetro crítico de los sistemas de procesamiento masivo de transacciones (reservas de vuelos, servicios bancarios.
 - o Debe medirse en “condiciones pico” del sistema.
- Optimizar el aprovechamiento del espacio: Es decir, la cantidad de espacio ocupado por ficheros de la BD y sus estructuras de acceso.
- Proporcionar la máxima seguridad.
- Evitar las reorganizaciones costosas en tiempo y recursos.

2.2. FACTORES QUE INFLUYEN EN EL DISEÑO FÍSICO

Un esquema lógico de BD tiene muchos esquemas físicos posibles en un determinado SGBD. Entonces, ¿cuál es el más apropiado? ¿En qué podemos basarnos para decidimos por uno u otro? Es imposible analizar el rendimiento ni tomar decisiones de diseño físico sin saber qué uso se le va a dar a la base de datos. Para ello es necesario analizar los siguientes puntos:

1. Consultas y transacciones que se espera ejecutar.
2. Frecuencia esperada de consultas y transacciones.
3. Restricciones de tiempo sobre consultas y transacciones.
4. Frecuencia esperada de operaciones de actualización.
5. Restricciones de unicidad sobre los atributos

Para analizar adecuadamente estos factores veamos detalladamente cuáles son las fases del diseño físico.

2.3. FASES DEL DISEÑO FÍSICO

El diseño físico se divide de cuatro fases, cada una de ellas compuesta por una serie de pasos:

1. Traducir el esquema lógico para el SGBD específico.

Los objetivos de esta fase son:

- Diseñar las relaciones base para el SGBD específico.
- Diseñar las reglas de negocio para el SGBD específico.

En esta fase se incorporan acciones como:

- Determinar si el sistema soporta la definición de claves primarias, claves ajenas y claves alternativas.
- Determinar si el sistema soporta la definición de datos requeridos (es decir, si se pueden definir atributos como no nulos).
- Si el sistema soporta la definición de dominios.
- Si el sistema soporta la definición de restricciones o aserciones de usuario.
- Cómo se crean las tablas.

2. Diseñar la representación física.

En este caso, el diseñador debe saber cómo interactúan los dispositivos involucrados y como esto afecta a las prestaciones: memoria principal, CPU, entrada/salida a disco, red...

Además, debe identificarse la forma y estado de uso de las estructuras esenciales para el manejo óptimo de la base de datos, lo cual implica:

- Analizar las transacciones.
- Escoger las organizaciones de ficheros.
- Escoger los índices secundarios.



- Considerar la introducción de redundancias controladas.
- Estimar la necesidad de espacio en disco.

3. Diseñar los mecanismos de seguridad.

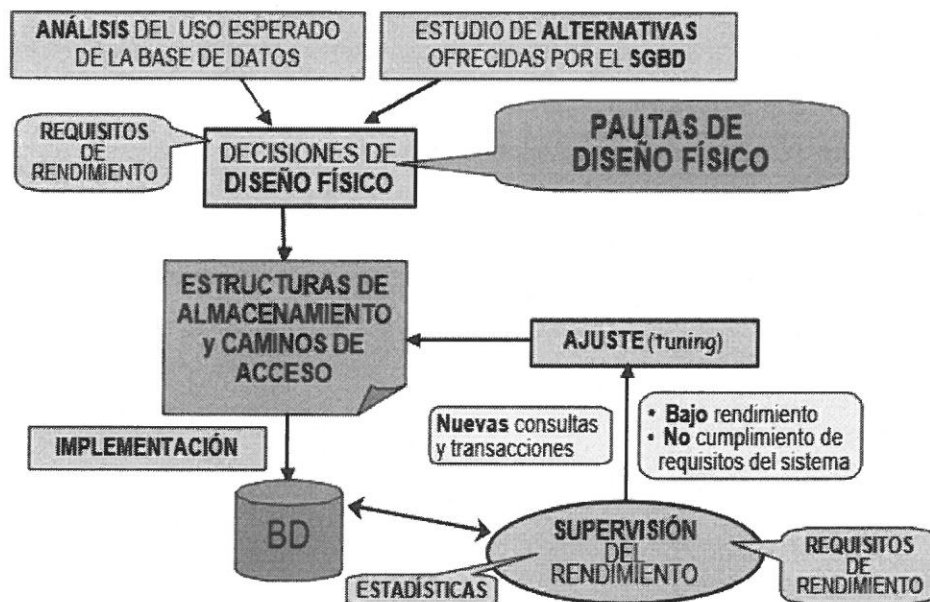
Requiere los siguientes pasos y actuaciones:

- Diseñar las vistas de los usuarios: CREATE VIEW.
- Diseñar las reglas de acceso: GRANT, REVOKE, triggers, etc.

4. Monitorizar y afinar el sistema.

Una vez implementado el esquema físico de la base de datos, se debe poner en marcha para observar sus prestaciones. Si estas no son las deseadas, el esquema deberá cambiar para intentar satisfacerlas. Una vez afinado el esquema, no permanecerá estático, ya que tendrá que ir cambiando conforme lo requieran los nuevos requisitos de los usuarios. Los SGBD proporcionan herramientas para monitorizar el sistema y conocer en cada instante su rendimiento.

En el siguiente esquema podemos resumir el diseño físico de datos:



3. DISEÑO FÍSICO DE FUNCIONES. CRITERIOS DE DISEÑO

La etapa del diseño físico es la etapa más cercana al código, por lo que siempre estará supeditada al tipo de código que vayamos a crear. Habitualmente se divide la implementación de un programa en componentes más sencillos que puedan ser desarrollados independientemente. Estos componentes tendrán distintos nombres en función del tipo de programación que se vaya a utilizar: objetos, funciones, módulos, procedimientos, etc. pero todos ellos tienen por objeto realizar esta división en componentes más sencillos. Cada componente está compuesto habitualmente por:

- Una sección de especificación que incluye el nombre del componente y la descripción de la interfaz de éste con el resto de la aplicación.
- Una sección de implementación que incluye el código propio del componente.
- Un mecanismo de activación que permite que el componente sea llamado desde cualquier otro punto del programa, se ejecute y devuelva el control.
- Comentarios con el objetivo de aclarar el diseño y funcionamiento del mismo.

Esta composición favorece que un componente tenga las siguientes características:

- Encapsula sus funciones y sus datos.
- Se define en base a su interacción con otros componentes.
- Es reutilizable entre distintas aplicaciones.
- Puede verse como una caja negra.
- Puede contener otros componentes.

3.1. CRITERIOS DE DISEÑO

Los criterios más importantes que se seguirán en este diseño físico serán:

- La modularización de las funcionalidades del sistema en componentes.
- La máxima cohesión de los componentes.
- El mínimo acoplamiento entre ellos.
- El nivel de granularidad, es decir, el nivel de detalle al que se llegará en la descomposición de los componentes. El tamaño de cada componente dependerá de su funcionalidad.
- La agregación y composición: El uso de técnicas de agregación y composición favorece la reutilización del código.
- Considerar la usabilidad del sistema como un requisito inherente al mismo.
- Diseño del manejo de errores y pruebas de eventos:
 - o Validar los parámetros a la entrada.
 - o Proteger recursos críticos.
 - o Proteger datos importantes.
 - o Debugging.
 - o Protección integral de transacciones de negocio.
- El estudio de las interfaces con el resto de componentes.

3.2. FASES DEL DISEÑO FÍSICO

El diseño físico de funciones comprende las siguientes tareas:

1. Definir los componentes.

Identificar cada uno de los componentes que formarán parte del sistema, las responsabilidades de cada uno de ellos y los servicios que ofrecerán (definidos por medio de sus interfaces).



2. Especificar las interfaces.

Especificar las interfaces de los componentes, identificando las interacciones que se establecen entre ellos y las entradas y salidas necesarias para establecer esa interacción.

3. Refinar el empaquetamiento y distribución en componentes.

Determinar cómo se organizarán entre ellos y cuál será su ubicación en el hardware y la ubicación del hardware en la red.

4. Documentar los componentes.

Documentar adecuadamente para que pueda comprenderse la funcionalidad del módulo sin necesidad de recurrir al análisis.

5. Distribuir los repositorios físicos de datos.

La distribución de datos puede establecerse en datos centralizados, partición de datos, extracto de datos y réplica de datos.

6. Examinar la tolerancia a fallos y la recuperación de errores.

Se realiza mediante pruebas de estrés. El sistema será tolerante a fallos si contiene componentes replicados o es redundante. La redundancia se puede implementar de tres formas diferentes: redundancia de información, redundancia de tiempo y redundancia física.

7. Validar el diseño físico.

La prueba de software es un elemento crítico para obtener la garantía de que las aplicaciones cumplen con los niveles de calidad requeridos y verificar las especificaciones del diseño. Una estrategia de pruebas debería incluir pruebas unitarias, pruebas de integración, pruebas del sistema, pruebas de aceptación y pruebas de regresión.

4. DOCUMENTACIÓN

La mayoría de la documentación técnica del proceso ya debería haber sido creada antes de llegar al diseño físico ya que esta es la última fase del proceso de diseño del software. Aún así en esta fase se crearán algunos documentos nuevos y se detallarán determinados aspectos de algunos documentos creados anteriormente. Entre esos documentos podemos destacar:

- **Diccionario de datos:** Este es un documento “vivo” que deberá ser actualizado durante todas las fases de diseño del sistema por lo que todas las modificaciones llevadas a cabo durante el diseño físico se trasladarán al diccionario de datos.
- **Documento de diseño técnico:** Este documento contemplará:
 - Diseño de la arquitectura del sistema.
 - Diseño físico de los datos.
 - Entorno tecnológico del sistema.
 - Plan de desarrollo e implantación.
 - Control de calidad.
- **Plan de pruebas:** Refleja la planificación de las pruebas y sirve de guía para la ejecución de las pruebas del software. Describe el alcance, recursos dedicados y la planificación temporal de las actividades de prueba.
- **Guía de instalación:** Describe el proceso a llevar a cabo para desplegar el sistema en un nuevo entorno. La guía de instalación será el documento que seguirán los equipos de mantenimiento para implantar la aplicación en los distintos entornos de la organización.

5. CONCLUSIÓN

La importancia del tema que nos concierne es obvia debido a que se centra en la fase del diseño físico llegando a la finalización del producto correspondiente y será cuando comprobemos si todo el trabajo realizado en todas las fases anteriores ha sido satisfactorio y ha dado los resultados esperados. Si todo el trabajo en fases anteriores ha sido adecuado y ha sido bien documentado nuestra labor realizando el diseño físico será mucho más sencilla y adecuada ya que para esta fase debemos partir de todo lo realizado anteriormente.

Se ha visto durante el tema cuáles son los criterios de diseño más relevantes para el diseño físico de datos y de funciones, así como la documentación que se crea en esta fase. Todos estos conceptos son de especial importancia dominarlos para acabar de realizar la construcción de nuestro sistema informático con éxito.

6. BIBLIOGRAFÍA

- Pressman, R. S. **Ingeniería del Software: Un enfoque práctico**. Editorial McGraw-Hill
- Sommerville, I. **Ingeniería de software**. Editorial Addison-Wesley
- <https://manuel.cillero.es/doc/metrica-3/procesos-principales/dsi/>