

PROGRAMA DE EMPREGO

FORMAWEB IV

DATA:	Marzo 2022
CURSO:	PROGRAMA DE EMPREGO FORMAWEB IV
MÓDULO	MF0966_3. Consulta e manipulación de información contida en xestores de datos
UNIDADE FORMATIVA	UF2214: Implementación e uso dunha base de datos
NºEXP:	36/00004/2021

PROGRAMA DE EMPREGO

FORMAWEB IV

Deseño de DB

Introdución

1. Proceso de deseño de bases de datos

2. Fases do deseño dunha base de datos

 2.1. Fase 1. Recollida e análise de requirimentos

 2.1.1. Recollida de requirimentos

 2.1.2. Estruturación e perfeccionamento dos requirimentos

 2.1.3. Formalización dos requirimentos

 2.2. Fase 2. Deseño conceptual

 2.2.1. O modelo ER

 2.2.2. A linguaxe de modelaxe unificada

 2.3. Fase 3. Deseño lóxico

 2.3.1. Reconsideracións do modelo conceptual

 2.3.2. Transformación do modelo conceptual no modelo lóxico

 2.3.3. Normalización

 2.4. Fase 4. Deseño físico

 2.4.1. O nivel físico e o nivel virtual

 2.4.2. Transformación do modelo lóxico en modelo físico

 2.5. Fase 5. Implantación e optimización

 2.5.1. Procesamento e optimización de consultas

 2.5.2. Ver procesamento

 2.5.3. Xestión da seguridade

Resumo

OBXECTIVOS

1. Comprender en que consiste o deseño de bases de datos e cales son os seus cometidos.

2. Coñecer as distintas etapas que conforman o proceso de deseño dunha base de datos.

Este é só un texto introdutorio ao deseño de bases de datos, no que se presenta unha visión xeral de todo o proceso de deseño. Aínda que o texto céñtrase no deseño de bases de datos relacionais, as fases pódense utilizar no deseño de bases de datos de calquera tipo (orientado a obxectos, relacional orientado a obxectos, etc.).

Deseño de DB

Introdución

O deseño de bases de datos é un proceso complexo que permite obter unha implementación de bases de datos a partir dos requisitos iniciais dos usuarios do sistema de información. Este proceso guía ao deseñador de bases de datos a través de varias etapas co obxectivo de segmentar un problema de considerable complexidade en diferentes subproblemas de menor complexidade.

Cada un dos subproblemas identificados corresponde a unha das etapas do **proceso de deseño da base de datos**.

1. Proceso de deseño de bases de datos

O proceso de deseño de bases de datos consiste en definir a estrutura lóxica e física dunha ou varias bases de datos para satisfacer as necesidades de información dos usuarios e para un determinado conxunto de aplicacións.

Mediante un proceso de deseño de bases de datos pódese decidir as táboas e relacións que debe ter unha determinada base de datos, os atributos das diferentes táboas, as claves primarias e as claves foráneas que se deben declarar en cada táboa, etc. En definitiva, se define que información é necesaria para dito sistema de información e como se relaciona esta información entre si. A demais de definir a información necesaria, tamén hai que ter en conta a forma de almacenar dita información para que os sistemas de información poidan funcionar de forma eficiente. Todas estas tarefas forman parte do proceso de deseño da base de datos. Para poder tomar estas decisións da mellor maneira, hai que ter en conta as necesidades de información dos usuarios en relación a un conxunto específico de aplicacións.

Por exemplo, supón que quere crear unha base de datos para apoiar o proceso de retirada de diñeiro dos caixeiros automáticos dun banco determinado. As primeiras fases do deseño da base de datos encargaránse de garantir que a base de datos conteña a información relevante: sobre as tarxetas de crédito, as súas contas vinculadas e o seu saldo. As últimas etapas do deseño da base de datos permitirán que se implemente de forma que poida responder á pregunta de se unha conta ten un saldo suficiente en menos de 2 segundos, independentemente de que existan millóns de contas no banco. .

Polo tanto, **o deseño dunha base de datos é o proceso no que se define a estrutura dos datos que debe ter a base de datos dun determinado sistema de información e como estes datos deben ser almacenados e xestionados para permitir unha explotación eficiente dos mesmos polos sistemas de información.**

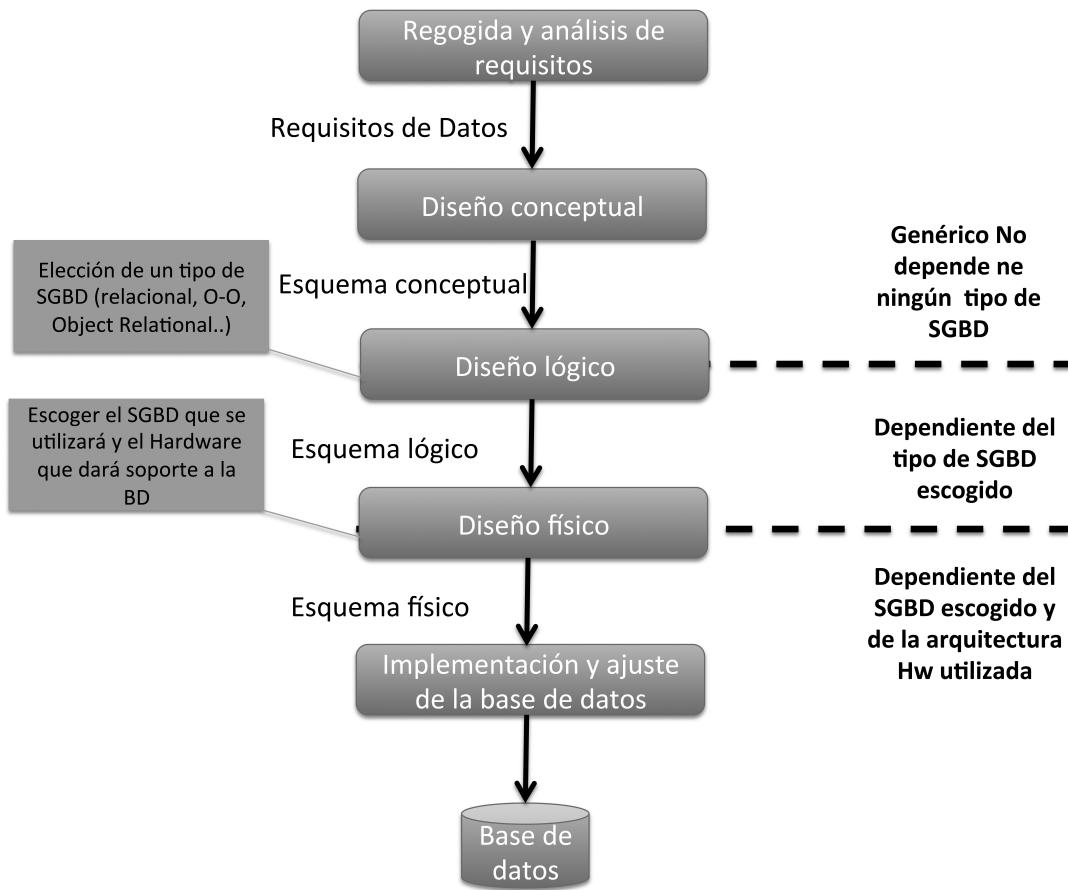
Os requisitos que debe cumplir un sistema de información e a complexidade da información que nel se presenta fan que o deseño dunha base de datos sexa un proceso complicado. Para simplificar este proceso, é moi recomendable utilizar a estratexia *divide e vence*.

A estratexia "divide and conquer" propón resolver un problema complexo subdividindo nun conxunto de problemas máis sinxelos onde a resolución dos diferentes subproblemas implica resolver o problema inicial.

Deste xeito, un proceso de certa complexidade descomponse en diferentes procesos de menor compleixade. Se aplicamos este concepto, pódense recoñecer as diferentes etapas do deseño da base de datos. Estas son secuenciais e o resultado de cada unha das serve como punto de partida para a seguinte etapa. O resultado da última etapa será a implantación da nosa base de datos.

A figura 1 mostra as diferentes etapas do deseño da base de datos.

Figura 1. Etapas do deseño da base de datos



En primeiro lugar, temos a **recolección e análise de requisitos**. Esta etapa debería permitir coñecer os requisitos de datos e as limitacións do problema. Para obter esta información será necesario manter conversas cos distintos usuarios da futura base de datos e das aplicacións que estean relacionadas con ela. Só se se cruzan os requisitos dos distintos perfís de usuario será posible establecer un marco completo de requisitos e as restricións dos datos relacionados coa futura base de datos.

A continuación, comeza o **deseño conceptual**. Nesta etapa, elabórase un esquema conceptual de alto nivel a partir das especificacións e requisitos obtidos na etapa anterior. Neste proceso cómpre extraer as necesidades e requisitos do problema e sintetizalos nun modelo visual de tal forma que permita representar os datos e as restricións dos conceptos que se van modelar no sistema de información. Este modelo chámase *esquema conceptual*.

Ata esta fase de deseño de bases de datos aínda non foi necesario escoller o tipo de base de datos a utilizar (relacional, orientada a obxectos, documental, etc.) nin o sistema de xestión de bases de datos (SXBD). utilizado ou a linguaaxe específica coa que se vai implementar a base de datos.

Un sistema de xestión de bases de datos (DBMS^[1]) é un tipo específico de software que serve de interface entre a base de datos, as aplicacións que a usan e o usuario.

No momento en que comeza a terceira etapa do proceso de deseño, o **deseño lóxico**, é necesario determinar o tipo de bases de datos que se vai utilizar. Aínda non é necesario escoller un DBMS específico, senón o tipo de bases de datos. Por *tipos de bases de datos* entendemos os distintos grupos de bases de datos segundo o modelo de datos que aplican. Algúns exemplos de tipos de bases de datos que se poden considerar son os seguintes: Relacional, Orientada a Obxectos, Relacional a Obxectos, NoSQL, Xeográfica, XML e Multidimensional. É importante ter en conta que neste paso NON estás elixindo un produto de base de datos específico, como Oracle, Sql Server, MySQL, PostgreSQL, VoldDB ou MariaDB, senón un tipo de base de datos. Nesta fase, o esquema conceptual convértese nun esquema lóxico axeitado ao tipo de bases de datos que se pretenden utilizar.

Neste punto, e antes de iniciar a fase de **deseño físico**, débese escoller un DBMS específico no que se vai implementar a base de datos. A fase de deseño físico adapta o esquema lóxico ás necesidades específicas dun DBMS concreto e, posteriormente, axusta algúns parámetros para o correcto funcionamento da base de datos. Por *base de datos concreta* ou *SGDB concreta* entendemos unha aplicación concreta das bases de datos. No caso das bases de datos relacionais, exemplos de DBMS específicos son Oracle Database, Mysql, SQL Server ou IBM Informix, entre outros. Tamén hai que ter en conta a arquitectura hardware que vai soportar a base de datos e o uso previsto da base de datos, xa que non é o mesmo implantar unha base de datos nun ordenador persoal que nunha *nube* de equipos con discos RAID, ou crear un escritorio. base de datos que é unha base de datos multiusuario con acceso 7x24.

Finalmente, a última etapa é a **implantación e optimización** da base de datos. Esta etapa permite cargar os datos e posteriormente axustar algúns parámetros do modelo físico e optimizar o rendemento da base de datos.

Estas fases de deseño non teñen que ser seguidas de xeito estritamente secuencial, e en moitos casos é habitual refacer o deseño da etapa anterior en función das necesidades detectadas en etapas posteriores. Estes **bucles de retroalimentación** son comúns e permiten refinar os deseños das distintas etapas dun xeito iterativo.

O proceso que se mostra na Figura 1 baséase no modelo **de deseño baseado en datos**. Este modelo céntrase no deseño de contedores de información e na estrutura da base de datos. Paralelo a este modelo está o modelo **de deseño orientado a procesos**, que se centra nas aplicacións de bases de datos para determinar os datos e como se usan. Tradicionalmente, o deseño de aplicacións baseouse neste segundo modelo, pero cada vez está más claro que ambas actividades son paralelas e están estreitamente relacionadas entre si. As ferramentas de deseño de bases de datos e aplicacións están cada vez más combinadas.

[1]: DBMS é o acrónimo de Database Management System ou Sistema de Xestión de Bases de datos.

2. Fases do deseño dunha base de datos

A continuación veremos con un pouco máis de detalle as fases que conforman o proceso de deseño dunha base de datos.

2.1. Fase 1. Recollida e análise de requisitos

A primeira fase no deseño dunha base de datos consiste en coñecer e analizar polo miúdo as expectativas, necesidades e obxectivos dos futuros usuarios da base de datos. Este proceso chámase *recollida e análise de requisitos*.

A fase de recollida e análise de requisitos pódese dividir en tres subfases secuenciais: a recollida de requisitos; a estruturación e perfeccionamento dos requisitos, e a formalización dos requisitos.

2.1.1. Recollida de requisitos

Para determinar os requisitos, primeiro é necesario establecer os actores do sistema de información que interactuarán coa base de datos. Isto inclúe usuarios e aplicacións, sexan novos ou non. Normalmente, un grupo de analistas é o encargado de facer a análise de requisitos e, moi probablemente, esta análise adoita ser informal, incompleta e mesmo inconsistente nalgún momento. Polo tanto, hai que dedicar moito esforzo a traballar esta información e convertela nunha especificación que os deseñadores de bases de datos poidan usar para modelar e implementar o sistema de información.

Nesta fase, non só é necesario recoller e analizar os requisitos relativos á estrutura ou forma da información (tipos de datos e relacións entre elementos de datos), senón que tamén hai que captar e analizar calquera tipo de esixencia. No caso particular da base de datos, os requisitos que os usuarios esperan da base de datos deben ser recollidos, analizados e documentados. Isto inclúe identificar os procesos que se deben executar na base de datos e estimar a frecuencia con que se executarán, as restricións que se deben implementar nos datos, as restricións ao rendemento do sistema de información, as restricións relacionadas coa implantación (tanto en termos de hardware e software), requisitos de seguridade ou rendemento (por exemplo, tempo de resposta) e volume esperado de datos, entre outros.

Algunhas das actividades más habituais desta fase son:

- Identificar os grupos de usuarios e as principais áreas de aplicación que utilizarán a base de datos e que se verán afectados directa ou indirectamente pola mesma. Dentro de cada grupo, seleccionar usuarios clave e formar comités para levar a cabo a recollida e especificación de requisitos.
- Estudar e analizar a documentación existente relacionada coas aplicacións en uso.
- Estudar o entorno actual e o uso que se pretende da información. Isto inclúe o estudo das entradas, fluxos e saídas de información, así como das frecuencias e usos das diferentes tarefas dentro do sistema de información.
- Realizar entrevistas e enquisas aos futuros usuarios para que expresen a súa opinión e prioridades sobre o novo sistema de información.

Ao final desta etapa, haberá unha lista preliminar e non formalizada dos requisitos do sistema a implementar.

2.1.2. Estruturación e perfeccionamento dos requisitos

Hai que ter en conta que algúns destes requisitos moi probablemente cambiarán durante o proceso de deseño e que é necesario estar atentos e en contacto permanente cos usuarios da base de datos para detectar posibles problemas. É unha boa práctica incorporar usuarios de bases de datos durante o proceso de desenvolvemento, xa que isto aumenta o seu nivel de implicación e satisfacción. Existen algunas metodoloxías propostas para a recollida e análise de

requirimentos baseadas en que desenvolvedores traballan xunto cos usuarios de bases de datos, como o Deseño de aplicacións convxuntas (JAD^[2]).

[2]: JAD é o acrónimo da expresión inglesa *joint application design - deseño de aplicacións convxuntas*.

2.1.3. Formalización dos requisitos

O seguinte paso é converter os requisitos nun formato estruturado utilizando técnicas de especificación de requisitos como análise orientada a obxectos (OOA^[3]), diagramas de fluxo de datos (DFD^[4]) ou notación Z^[5]. Estas técnicas usan diferentes tipos de recursos (diagramas, texto, táboas, gráficos, diagramas de decisión, etc.) para organizar e representar os requisitos de forma clara.

Esta fase pode representar un custo importante no proceso de deseño da base de datos, pero é moi importante e pode ser decisiva para o éxito ou o fracaso do sistema de información. Detectar e corrixir os erros ou problemas nas fases iniciais do proxecto é moito menos custoso que arrastrar os erros ata as fases finais, cando corrixilos terá un custo moi superior. A satisfacción do usuario final virá determinada pola capacidade de recoller e captar as súas necesidades e implementalas correctamente na solución final. Ademais, se son comprensibles polo cliente, os requisitos formalizados poden ser un bo documento de compromiso entre o cliente e o deseñador da base de datos, xa que contén toda a información sobre o que hai que facer e como facelo.

[3]: OOA é o acrónimo da expresión inglesa *object oriented analysis - análise orientada a obxectos* .

[4]: DFD é o acrónimo da expresión inglesa *data flow diagrams - diagramas de fluxo de datos*.

[5]: Notación Z é unha linguaxe formal utilizada na enxeñaría de software.

2.2. Fase 2. Deseño conceptual

A fase de *deseño conceptual* pretende crear un esquema conceptual de alto nivel e independente da tecnoloxía baseado nos requisitos, especificacións e restricións que se recollerón na fase anterior.

Esta fase baséase na recollida e análise de requisitos obtidos na fase anterior e ten como obxectivo deseñar un esquema conceptual da base de datos coherente cos requisitos, especificacións e restricións impostas polo problema a resolver. Isto é representar a información que necesita a base de datos, xunto coas súas limitacións.

Este esquema conceptual ou representación gráfica describe o sistema de información e como vai a levar a cabo as súas funcións. Para crear un esquema conceptual, os deseñadores deben ter un bo coñecemento do sistema de información (de aí a necesidade de ter unha boa relación de requisitos formalizados) e unha gran capacidade de abstracción.

Os esquemas conceptuais son modelos de alto nível que describen información e, polo tanto, non inclúen detalles de implementación. Un esquema conceptual, ademais, debe servir de referencia para comprobar que todos os requisitos foron agrupados e sirve tamén para comprobar que non existe conflito entre eles. De feito, dependendo da linguaxe empregada para modelar o esquema conceptual, podemos atopar ferramentas que nos permiten analizar a súa validez e calidade, como a ferramenta USE, que permite avaliar se un esquema UML é correcto e se a súa integridade as restricións son satisfacibles.

Un **esquema conceptual** é unha descripción concisa dos requisitos de datos e as súas relacións, fácil de entender e sen detalles de implementación

Nesta fase de deseño aínda non se ten en conta o tipo de base de datos a utilizar. E, polo tanto, nin o DBMS nin a linguaxe específica de implementación da base de datos. Nesta fase concentrarémonos na estrutura da información, sen resolver polo momento cuestións relacionadas coa tecnoloxía.

Existen varios modelos de datos de alto nivel que permiten modelar os requisitos, especificacións e restricións que se obtiveron na primeira fase do deseño da base de datos. Estes modelos dispoñen normalmente de linguaxes gráficas para facilitar a representación e comprensión dos seus esquemas conceptuais.

2.2.1. O modelo ER

Un dos máis coñecidos e utilizados é o modelo entidade-relación, que abreviaremos como **modelo ER**^[6]. Este modelo é un dos más empregados no deseño conceptual de aplicacións de bases de datos, principalmente pola súa sinxeleza e facilidade de uso.

Os principais elementos incluídos no modelo son os tipos de entidades, os atributos e os tipos de relacións entre as entidades. O obxectivo principal do modelo ER é permitir aos deseñadores reflectir nun modelo conceptual os requirimentos do mundo real que son de interese para o problema. O modelo ER facilita o deseño conceptual dunha base de datos e, como se dixo, é aplicable ao deseño de calquera tipo de base de datos.

[6]: En inglés chámase modelo *entity-relationship*. Hai quen o traduce como *modelo entidade-relación* ou tamén como *modelo entidade-interrelación*. Ambos conceptos refírense ao mesmo modelo.

2.2.2. A linguaxe de modelaxe unificada

A Unified Modeling Language (UML^[7]) é unha linguaxe gráfica deseñada para especificar, visualizar, modificar, construir e documentar un sistema. A linguaxe UML incorpora un gran número de diagramas que permiten representar o modelo dun sistema desde diferentes perspectivas. En relación ao deseño conceptual das bases de datos, é de especial interese o diagrama de clases, que permite representar información do dominio do sistema de información que se vai implantar. Os diagramas de clases son diagramas estáticos que describen a estrutura dun sistema baseándose nas clases ou tipos de entidades do sistema, os seus atributos e as asociacións ou tipos de relacións que se establecen entre eles. Estes diagramas mostraron unha excelente capacidade de modelado de datos. Por este motivo, son cada vez más importantes no deseño conceptual de bases de datos.

[7]: Unified Modeling Language é unha linguaxe de propósito xeral para modelar sistemas de software. Mantido actualmente polo Object Management Group (OMG).

2.3. Fase 3. Deseño lóxico

Antes da fase de deseño lóxico, debese escoller un tipo de base de datos. Noutras palabras, aínda non tes que escoller un DBMS específico, senón que simplemente selecciona o tipo de base de datos que queres implementar. É importante ter claro que o tipo de base de datos determina o esquema de deseño lóxico. Unha vez escollido o tipo de DBMS onde quere implementar a base de datos, pode comezar a fase de deseño lóxico.

Na fase de **deseño lóxico**, o modelo conceptual, independente do modelo de datos a utilizar na base de datos, transfórmase nun modelo lóxico dependente do modelo de datos (ou tipo de SGBD) no que se vai implantar.

Esta etapa iníciase co deseño conceptual desenvolvido no paso anterior e obtense un deseño lóxico da futura base de datos. A transformación traducirá o modelo tendo en conta o tipo de DBMS no que se vai implementar a base de datos. Por exemplo, se quere crear a base de datos nun sistema relacional, esta etapa obterá un conxunto de relacións cos seus atributos, chaves primarias e chaves foráneas correspondentes. Mientras que se a base de datos debe crearse nun sistema orientado a obxectos, identificaranse as diferentes clases que se van crear, os seus atributos, relacións, restricións de integridade e a xerarquía de especializacións entre clases e posiblemente entre clases.

Nesta etapa centrámonos nas cuestións tecnolóxicas relacionadas co modelo de base de datos, asumindo que na etapa anterior xa resolvemos o problema da estruturación da información dende o punto de vista conceptual.

Polo tanto, o resultado desta etapa será un modelo lóxico da estrutura da información. Xeralmente, cando este modelo lóxico se refire a un DBMS relacional, denomínase *modelo relacional*.

No deseño lóxico pódense apreciar tres subfases ou partes independentes, que se aplican secuencialmente para transformar o modelo conceptual obtido na fase anterior nun modelo lóxico que será o resultado desta fase. En primeiro lugar, na subfase denominada *reconsideracións do modelo conceptual*, revisamos o modelo conceptual para asegurarnos de que está libre de algúns erros tipográficos identificables. A continuación, prodúcese a transformación do modelo conceptual no modelo lóxico e, finalmente, aplicamos a teoría da normalización ao modelo lóxico.

2.3.1. Reconsideracións do modelo conceptual

Nesta primeira parte realiza unha análise en profundidade do modelo conceptual obtido na fase anterior, coa intención de detectar e corrixir algúns erros que adoitan producirse nos modelos conceptuais e que deben ser detectados e reparados canto antes para evitar a súa propagación en fases posteriores. Estes erros chámense *trampas de deseño*. É unha boa idea ter en conta cada un destes erros e asegurarse de que o modelo conceptual que quere transformar está libre deles antes de continuar co deseño lóxico.

2.3.2. Transformación do modelo conceptual no modelo lóxico

No deseño lóxico de bases de datos relacionais - no que estamos a centrarnos-, o modelo lóxico xerado será aplicable a calquera base de datos relacional. Partindo do resultado da etapa de deseño conceptual, o esquema conceptual reescríbese segundo modelo de datos do tipo de SGBD elixido. Por exemplo, no caso de querer utilizar un DBMS relacional, o esquema conceptual reescríbese mediante táboas, atributos, chaves primarias e chaves foráneas. Expresado a través dun diagrama de clases UML. No proceso se leva a cabo a transformación dos distintos elementos que conforman o modelo conceptual en elementos que conforman o modelo lóxico ou, más concretamente, o modelo relacional.

O modelo relacional é o modelo lóxico específico das bases de datos relacionais.

2.3.3. Normalización

A teoría da normalización aplica a **teoría de conxuntos**, a **lóxica** e a **álgebra relacional** para formalizar un conxunto de ideas sinxelas que guían un bo deseño de bases de datos relacionais.

A teoría da normalización utiliza formas normais (FN) para recoñecer casos nos que non se aplican os criterios de bo deseño. Unha relación está nunha forma normal dada se satisface un conxunto de restricións específicas que son propias desta forma normal. A violación destas restricións fai que a relación teña un conxunto de anomalías de actualización indeseables e redundancias. As formas normais son declarativas, é dicir, cada forma normal indica as restricións que se deben cumplir, pero non describe ningún procedemento para conseguiilo.

2.4. Fase 4. Deseño físico

Antes da fase de deseño físico, debe escollerse un DBMS específico. É necesario estudar os diferentes sistemas comerciais ou libres que hai no mercado e seleccionar un DBMS onde se poida implantar o sistema de información que se desenvolveu nas fases anteriores do proceso de deseño.

Os compoñentes físicos que componen cada DBMS son específicos. Os fabricantes utilizan diferentes estratexias e tecnoloxías para maximizar o rendemento dos seus sistemas de xestión de bases de datos. Neste nivel non existe un estándar e, polo tanto, haberá que adaptar o esquema lóxico obtido no paso anterior, tendo en conta as características de cada sistema de xestión. O deseñador debe considerar os aspectos físicos de implementación e eficiencia que dependen especificamente do SGBD elixido.

O **deseño físico** é unha fase do proceso de deseño da base de datos que adapta o esquema lóxico obtido na fase anterior ao SGBD específico que será utilizado polo sistema de información. Nesta fase terase tamén en conta unha estimación do uso esperado dos datos na base de datos, co fin de atopar unha configuración física da base de datos adaptada ao entorno onde se aloxará e que permita o almacenamento e o uso de datos cun rendemento adecuado.

Do anterior pódese extraer que o hardware onde se situará a base de datos e as características dos procesos de consulta e actualización da base de datos, as frecuencias de execución e os volumes previstos dos diferentes datos a almacenar tamén son relevantes na busca dunha solución eficiente.

2.4.1. O nivel físico e o nivel virtual

O estudo dos niveis físico e virtuais das bases de datos permítenos ver aspectos como as estruturas de almacenamento e as vías de acceso a través dos ficheiros da base de datos. Cada SGBD ofrece diferentes opcións para organizar ficheiros e rutas de acceso, sendo necesario que o deseñador coñeza a implantación concreta, para poder implementar o deseño físico do sistema de información dun xeito máis eficiente.

Tamén é importante coñecer as características dos procesos que consultan e actualizan a base de datos, como as frecuencias de execución e os volumes previstos dos diferentes datos que se van almacenar para conseguir un bo funcionamento da base de datos.

Algúns criterios importantes que poden ser útiles para escoller as opcións de deseño de bases de datos físicas inclúen os seguintes:

- **Tempo de resposta**. É o tempo que transcorre desde que se envía unha solicitude ao DBMS ata que devolve os datos de resposta. Unha parte importante deste tempo está baixo o control do DBMS e refírese ao tempo de acceso por parte do DBMS aos datos necesarios

para xerar a resposta. Outros aspectos non están controlados polo SGBD, como a planificación do sistema operativo ou os tempos de acceso aos medios físicos de almacenamento de datos, aínda que o modelo físico que definimos pode condicionar fortemente estes últimos, xa que nos permite decidir cal os discos deberían albergar determinados datos, de que forma (orientado a columna ou fila, ordenado ou non, etc.) e o tamaño das páxinas de datos, entre outros.

- **Uso do espazo** . É a cantidade de espazo en disco que usan os ficheiros de bases de datos e as estruturas de rutas de disco, incluíndo índices e outras rutas.
- **Rendemento** . É o número medio de transaccións que se poden procesar nun minuto de tempo. Este factor pode ser crítico para sistemas transaccionais, como compañías aéreas ou bancos.
- **Dispoñibilidade da base** de datos . Nalgúns casos, é importante asegurarse de que a base de datos será resistente a fallos e fallos. Por exemplo, no caso dun banco, débese garantir que os datos estarán accesibles 7x24 e que se nalgún momento a base de datos deixa de funcionar, o tempo de recuperación será mínimo. Hai que ter en conta que moitas empresas hoxe en día deben a súa existencia ao procesamento automatizado de datos e non poden funcionar sen el. Pense, por exemplo, cales serían os efectos se un banco non puidese acceder aos seus datos durante 2 días seguidos: operacións de oficinas inoperativas, caixeiros automáticos non operativos, non saber que contas correntes hai e con que saldos, etc.

Os compoñentes físicos que componen cada DBMS son específicos. Os fabricantes utilizan diferentes estratexias e tecnoloxías para maximizar o rendemento dos seus sistemas de xestión de bases de datos. Incluso a linguaxe utilizada para definir o modelo físico varía dun fabricante a outro. Isto débese a que non existe un estándar para definir as compilacións adecuadas para este nivel, a diferenza dos niveis anteriores. Por exemplo, o estándar SQL incorpora a definición de todos os compoñentes do deseño lóxico da base de datos, más as operacións de acceso aos datos da base de datos, xa sexa para consulta ou para inserción/actualización. Pola contra, non contén ningún elemento físico de deseño. Non obstante, existe unha gran semellanza entre as construcións empregadas polos distintos xestores, xa que todas permiten traballar cos mesmos elementos físicos de deseño. Por exemplo, no caso relacional, os elementos físicos de deseño utilizados son espazos de táboas , índices, vistas materializadas e particións.

Por iso, será necesario adaptar o esquema lóxico obtido no paso anterior, tendo en conta as características de cada sistema de xestión. O deseñador debe considerar cuestións de implementación física e eficiencia que dependen especificamente do DBMS elixido. Por exemplo, no caso de utilizar un DBMS relacional, partíramos das definicións da táboa (con toda a información relacionada, é dicir, atributos, chaves primarias, chaves foráneas e claves alternativas). A continuación, asociaríase cada elemento a un espacio axeitado a nivel virtual e, finalmente, cada espacio virtual cun arquivo físico, que constitúe o nivel físico do sistema de información.

2.4.2. Transformación do modelo lóxico en modelo físico

Neste paso, o modelo lóxico dunha base de datos transfórmase nun modelo físico, dependendo do SGBD elixido para implantar o sistema de información, das características específicas do hardware utilizado e do sistema operativo e software básico. Cada DBMS desenvolveu a súa propia linguaxe, adaptada polo propio construtor, para implementar o deseño físico da base de datos, segundo as características do entorno, e para obter o máximo rendemento do hardware, do sistema operativo e do propio xestor. . En realidade, podería considerarse unha extensión da linguaxe SQL estándar coas cláusulas propias que cada responsable precisa para definir os compoñentes do deseño físico. Porén, existe unha gran semellanza ou equivalencia entre unha parte importante dos distintos xestores.

O estándar SQL incorpora a definición de todos os compoñentes do deseño lóxico da base de datos. Pola contra, non contén ningún elemento do deseño físico.

Para transformar o deseño lóxico da base de datos nun DBMS específico, partimos das definicións da táboa (con toda a información relacionada, é dicir, atributos, claves primarias, claves foráneas e claves alternativas). A continuación, cada elemento está relacionado cun espazo axeitado a nivel virtual e finalmente cada espazo virtual está relacionado cun ficheiro físico, que constitúe o nivel físico do sistema de información.

2.5. Fase 5. Implantación e optimización

A última etapa é a implantación e optimización da base de datos.

A **fase de implantación e optimización** consiste na carga dos datos e despois axustar algúns parámetros relacionados co modelo físico da base de datos para optimizar o rendemento.

O obxectivo principal desta etapa é optimizar o rendemento da base de datos. En primeiro lugar hai que cargar os datos, xa que non é posible optimizar o acceso aos datos sen poder determinar o tamaño das táboas, os tipos de accesos e consultas, a súa frecuencia, etc.

Finalmente, tamén haberá que especificar os distintos roles dos usuarios e das aplicacións para poder determinar os permisos dos distintos grupos. O compoñente de xestión de seguridade e as vistas permitenche limitar o acceso e reducir así o risco de problemas derivados do acceso non autorizado.

2.5.1. Procesamento e optimización de consultas

A linguaxe SQL utilizada polas bases de datos relacionais é unha linguaxe declarativa. Isto implica que especifique o resultado que quere obter dunha consulta feita, en lugar de determinar o algoritmo ou o método a utilizar para obter o resultado. Polo tanto, é necesario comprender o conxunto de tarefas que realizan os SGBD relacionais para obter a resposta desexada.

Os SGBD relacionais avalían sistematicamente as posibles estratexias alternativas que se poidan presentar, co obxectivo de elixir aquela que se considere óptima. O procesamento de consultas inclúe todo o conxunto de actividades que realiza o SGBD, que teñen como obxectivo extraer información da base de datos para acadar a estratexia máis eficiente e proporcionar un mellor rendemento do sistema de información.

O procesamento de consultas pódese dividir nas seguintes catro etapas principais:

1) Descomposición da consulta . Tradución da consulta expresada en linguaxe SQL a unha representación interna baseada en álgebra relacional.

2) Optimización de consultas . O proceso de selección do plan de execución de consultas máis eficiente entre as moitas estratexias dispoñibles xeralmente para procesar unha consulta. A optimización de consulta pódese realizar en tres aspectos:

a) Optimización semántica en función das restriccions especificadas no esquema da base de datos.

b) Optimización sintáctica, que consiste en transformar heurísticamente a expresión relacional orixinal nunha equivalente moito más eficiente.

c) Optimización física, coa finalidade de escoller entre os distintos plans de avaliación –que poden ter custos distintos– o que resulte más eficiente. Para determinar o custo más eficiente, é necesario coñecer o custo de cada operación, que moitas veces depende de diferentes parámetros.

3) Xeración de código.

4) Execución da consulta.

É o proceso de optimización de consultas, é dicir, o tratamento que un DBMS dá ás consultas realizadas polos usuarios a través de SQL. Este punto é un dos más importantes que hai que ter en conta á hora de deseñar un DBMS relacional, xa que a opción elixida afecta directamente ao rendemento global do sistema.

A optimización de consultas é un aspecto moi importante a ter en conta ao deseñar e construír un DBMS relacional. As técnicas empregadas para optimizar as consultas condicionan o rendemento global do sistema, xa que determinan o tempo que precisa o sistema xestor para resolver as consultas realizadas polos usuarios.

2.5.2. Ver procesamento

Unha vista é unha táboa lóxica que permite acceder á información dunha ou máis táboas mediante unha consulta predefinida. Non contén información por si só, pero baséase en información doutras táboas. As vistas proporcionan mecanismos de seguridade e permiten ao deseñador da base de datos personalizar a vista que teñen os diferentes usuarios da base de datos.

O uso correcto das vistas é un aspecto fundamental para conseguir un bo deseño de base de datos, xa que nos permite ocultar os detalles das táboas e manter a visión do usuario independentemente da evolución da estrutura da táboa.

As vistas foron durante moito tempo un simple mecanismo de simplificación de consultas, pero hoxe en día teñen unha importancia primordial en varias áreas, como o deseño externo, os almacéns de datos (5) ou a computación distribuída.

(5) En inglés, *data warehouse* .

2.5.3. Xestión da seguridade

Por último, hai que ter en conta as técnicas empregadas para protexer a base de datos contra accesos non autorizados e os mecanismos de asignación e revogación de privilexios aos distintos usuarios. O compoñente de seguridade do DBMS encárgase destas e outras accións. Este compoñente cobra cada día máis importancia, xa que hoxe en día un gran número de ordenadores e outro tipo de dispositivos están interconectados e, polo tanto, calquera pode converterse en usuario, e posible atacante, dunha base de datos.

Facer un uso axeitado das vistas é outro aspecto clave da xestión da seguridade. As vistas son estruturas lóxicas que permiten acceder á información da base de datos desde unha consulta predefinida. Polo tanto, coas vistas pódense xerar diferentes visións dos datos adaptadas aos distintos usuarios, xa que podemos ocultar información a determinados usuarios e manter a visión do usuario independentemente da evolución da base de datos. Así, mediante vistas, sería posible facer que os traballadores do departamento de contabilidade dunha empresa só puidesen ver os datos de pagamento dun cliente (nome, NIF e conta corrente), mentres que os traballadores de loxística puidesen ver o seu enderezo e datos de pago. entrega (nome, persoa de contacto, teléfono, enderezo de entrega, etc.).

En moitas organizacións, a información é un activo intanxible de carácter sensible, que ten un valor moi importante. Para preservar esta información hai que protexer o sistema de información e coñecer as obrigas legais que deben cumplir.

Resumo

Acabamos de ver, dun xeito moi xeral, o proceso de deseño dunha base de datos e as diferentes etapas que o conforman.

O proceso de deseño dunha base de datos comeza coa **fase de recollida e análise de requisitos**, que permite recoller e centralizar as necesidades dos distintos grupos de usuarios e aplicacións. A partir desta análise, na segunda fase modelase un **esquema conceptual** que permite describir o modelo de datos de forma independente da tecnoloxía.

A seguinte etapa deste proceso é o **deseño lóxico**, que esixe ter escollido previamente o tipo de base de datos a empregar na implantación do sistema de información. O tipo de base de datos determina o modelo lóxico a desenvolver. Por exemplo, no caso de usar un tipo de base de datos relacional, o *modelo conceptual transformase nun modelo lóxico* específico para bases de datos relacionais chamado *modelo relacional*.

O **deseño físico** permitirá adaptar o modelo lóxico a un sistema de xestión de bases de datos específico (SGBD). Polo tanto, con anterioridade a este paso, débese escoller o SGBD específico que se utilizará para implantar o sistema de información. Nesta etapa, créase a estrutura física que almacenará os datos da base de datos.

Finalmente, a última etapa permite a **optimización da base de datos** e a **xestión da seguridade** relacionada cos usuarios e aplicacións da base de datos. Loxicamente, os datos terán que ser cargados previamente, xa que o tamaño das táboas, os tipos de consultas e as súas frecuencias son elementos importantes para optimizar o acceso aos datos por parte do sistema de xestión.

MRZ 2022



Xacobeo 21-22



CONCELLO
DE VIGO



OBRAOS DE
EMPLEO DE GALICIA



UNIÓN EUROPEA



XUNTA
DE GALICIA