

Altres temes de bases de dades

M. Elena Rodríguez González


P03/05053/02054

Índex

| | |
|--|----|
| Introducció | 5 |
| Objectius | 5 |
| 1. Optimització de consultes | 7 |
| 1.1. L'àlgebra relacional en el procesament de consultes | 8 |
| 1.2. El procés d'optimització de consultes | 11 |
| 2. Eines d'administració | 17 |
| 2.1. Eines d'administració central | 17 |
| 2.2. Eines d'edició i manipulació de les BD | 20 |
| 2.3. Resum de les eines d'administració | 21 |
| Resum | 22 |
| Activitats | 23 |
| Exercicis d'autoavaluació | 23 |
| Solucionari | 25 |
| Glossari | 27 |
| Bibliografia | 28 |

Altres temes de bases de dades

Introducció

En aquest mòdul didàctic tractem de temes relacionats amb la utilització dels SGBD relacionals. Concretament estudiem els aspectes següents: 

- 1) El **procés d'optimització de consultes**, és a dir, el tractament que dona un SGBD a les consultes fetes pels usuaris mitjançant SQL. Aquest és un dels punts més importants que cal tenir en compte quan es dissenya un SGBD relacional, ja que l'opció triada afecta directament el rendiment global del sistema.
- 2) Les **eines d'administració** més habituals que ofereixen els SGBD que hi ha al mercat per a facilitar la gestió de les BD que estan sota el seu control.

Objectius

En els materials didàctics inclosos en aquest mòdul l'estudiant trobarà les eines i els continguts necessaris per a assolir els objectius següents:

1. Entendre els objectius generals del processament de consultes.
2. Distingir els diferents tipus d'optimització que un SGBD relacional pot dur a terme.
3. Entendre les funcions de les eines d'administració dels SGBD.
4. Conèixer els diferents tipus d'eines d'administració que proporcionen els SGBD del mercat i saber quins serveis incorporen.

1. Optimització de consultes

L'optimització de consultes és un dels aspectes més importants que cal considerar quan es dissenya i es construeix un SGBD relacional, ja que les tècniques que s'utilitzen per a optimitzar consultes condicionen el rendiment global del sistema. En concret, afecta el temps que necessita l'SGBD per a respondre les consultes fetes pels usuaris.

Recordeu que en el Glossari trobareu el significat de les sigles que apareixen al llarg d'aquest mòdul didàctic.


El procés d'optimització de consultes està inclòs dins un procés més general que es coneix com *processament de consultes*. El **processament de consultes** consisteix en la transformació d'una consulta (expressada amb un llenguatge de consultes relacional, com ara l'SQL) en un conjunt d'instruccions de baix nivell. Aquest conjunt d'instruccions de baix nivell constitueix una estratègia per a accedir a les dades amb un consum de recursos mínim. Precisament, l'objectiu del procés d'optimització de consultes és trobar aquesta estratègia.

SQL

Recordeu que l'SQL (*Structured Query Language*) és el llenguatge de dades estàndard per a definir i manipular BD relacionals.

Els usuaris d'un SGBD relacional accedeixen a les BD gestionades per l'SGBD mitjançant consultes expressades en un llenguatge relacional, generalment en SQL. L'SGBD serà l'encarregat de processar aquestes consultes.

Vegeu les diferents modalitats de treball amb SQL a l'apartat 4 del mòdul didàctic "Introducció als sistemes de gestió de bases de dades" d'aquesta assignatura.

Per a simplificar l'estudi del problema suposarem que les consultes són sintàcticament i semànticament correctes. És a dir, les sentències d'SQL no contenen errors de sintaxi i les dades referenciades són correctes (les taules existeixen i les columnes també). 

Equivalència de termes

Com ja sabeu, les relacions, els atributs i els tuples són, en la terminologia del llenguatge SQL, les taules, les columnes i les files, respectivament.

Com ja sabeu, SQL es caracteritza principalment per ser un llenguatge declaratiu. La implicació principal d'aquest fet és que quan un usuari fa una consulta indica quines són les dades que vol i a quines taules es troben, però no diu com s'han de trobar aquestes dades ni en quin ordre s'han d'executar les operacions especificades a la consulta.

És missió de l'SGBD determinar com s'ha d'executar la consulta: ha de saber com estan emmagatzemades les dades i quins camins d'accés (índexs) s'hi han definit, i ha de determinar en quin ordre s'han d'executar les operacions que es demanen a la consulta independentment de la modalitat de treball amb SQL. Donada una consulta, tindrem moltes alternatives per a resoldre-la; l'SGBD haurà d'avaluar el cost de cadascuna i triar la que tingui el mínim; aquesta alternativa rep el nom de **pla d'accés de la consulta**.

Per a poder optimitzar una consulta, l'SGBD necessita fixar un ordre inicial de les operacions que inclou la consulta. Per tant, l'SGBD necessita passar d'un llenguatge declaratiu a un llenguatge procedimental. Aquest pas implica un procés de traducció, i el que fan els SGBD relacionals és traduir la consulta original (expressada en SQL) a una consulta formulada en àlgebra relacional.

A continuació veurem com es fa aquesta traducció dins el processament de consultes.

1.1. L'àlgebra relacional en el procesament de consultes

Com heu estudiat en altres assignatures de BD, per a especificar una consulta en àlgebra relacional és necessari seguir un o més passos que serveixen per a anar construint, mitjançant operacions de l'àlgebra relacional, noves relacions que contenen les dades que finalment responen a una consulta a partir de les relacions emmagatzemades a una BD. Es tracta d'un llenguatge procedimental, atès que els passos que formen la consulta descriuen un procediment.

Vegeu l'àlgebra relacional a l'apartat 5 del mòdul didàctic "El model relacional i l'àlgebra relacional" de l'assignatura *Bases de dades I*.

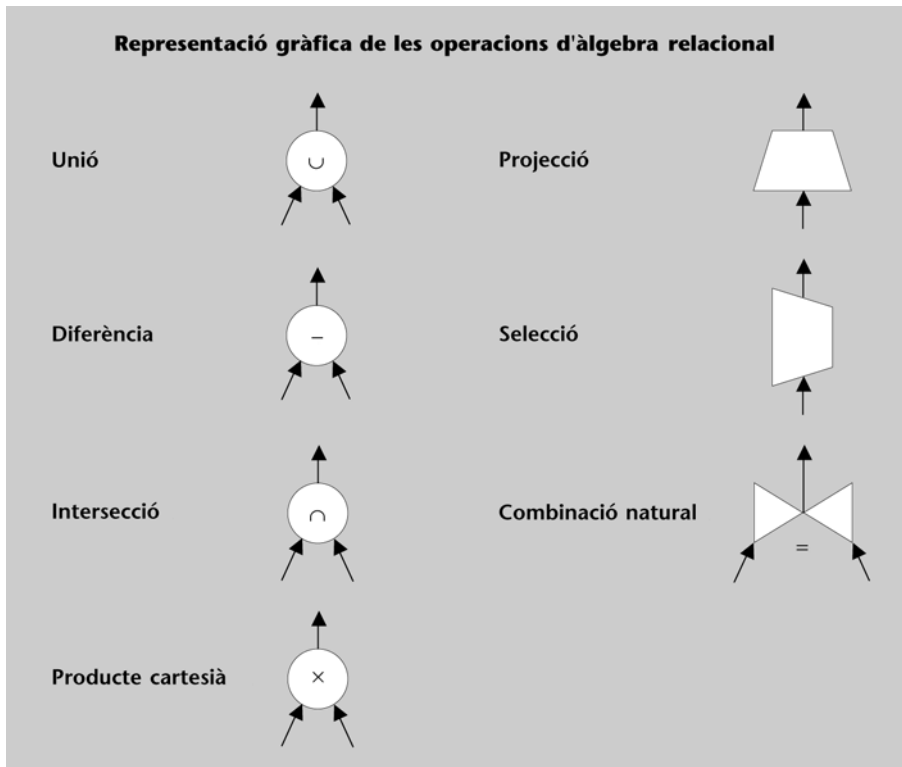


Donada una consulta formulada en llenguatge SQL, és possible trobar diverses consultes equivalents expressades en àlgebra relacional. Cadascuna d'aquestes consultes en àlgebra relacional es pot representar gràficament mitjançant una estructura en forma d'arbre que es coneix amb el nom d'*arbre sintàctic de la consulta*.

L'**arbre sintàctic d'una consulta** és una estructura en forma d'arbre que correspon a una expressió de l'àlgebra relacional en la qual les diferents parts de l'arbre representen els elements que presentem a continuació:

- 1) Les **fulles de l'arbre** representen les relacions de base (taules d'SQL) que intervenen a la consulta.
- 2) Els **nodes intermedis** són les operacions de l'àlgebra relacional que intervenen en la consulta original. L'aplicació de cadascuna d'aquestes operacions dóna lloc a una relació intermèdia.
- 3) L'**arrel de l'arbre** constitueix la resposta a la consulta que ha estat formulada.

Totes les operacions de l'àlgebra relacional es poden representar gràficament. A la figura següent hi trobareu la representació de cadascuna de les operacions que ja coneixeu:



A continuació presentem un exemple de traducció de consultes que il·lustra com es pot passar d'una consulta expressada en SQL a consultes equivalents en àlgebra relacional. 🗨️

Imaginem una BD d'una empresa dedicada al desenvolupament de sistemes d'informació. Aquesta empresa vol tenir constància (entre d'altres coses) de quins són els seus treballadors, quins projectes es desenvolupen en un moment determinat i quins empleats estan assignats en aquell moment als diferents projectes. A continuació mostrem els esquemes de les relacions que emmagatzemen aquestes dades:

- $EMPLEATS(\underline{num_empl}, \underline{nom_empl}, categoria_laboral, divisió, cap)$.
- $PROJECTES(\underline{num_proj}, \underline{nom_proj}, producte, duració)$.
- $ASSIGNACIONS(\underline{num_empl}, \underline{num_proj}, dedicació)$.

Els atributs subratllats són les claus primàries de cada relació. També tenim les claus foranes següents:

- L'atribut *cap* de la relació *EMPLEATS* és una clau forana que referencia la relació *EMPLEATS*.
- L'atribut *num_empl* de la relació *ASSIGNACIONS* és una clau forana que referencia la relació *EMPLEATS*.
- L'atribut *num_proj* de la relació *ASSIGNACIONS* és una clau forana que referencia la relació *PROJECTES*.

Suposem que les relacions han estat creades fent servir les sentències adients d'SQL i que s'han inserit diferents tuples a cadascuna de les relacions. Imaginem ara que en un moment donat un usuari vol saber dades dels empleats que dediquen més de 200 hores a un projecte. En concret, l'usuari vol trobar el nom i el número d'aquests empleats i els números dels projectes als quals estan assignats. Una possibilitat seria formular la consulta d'SQL següent:

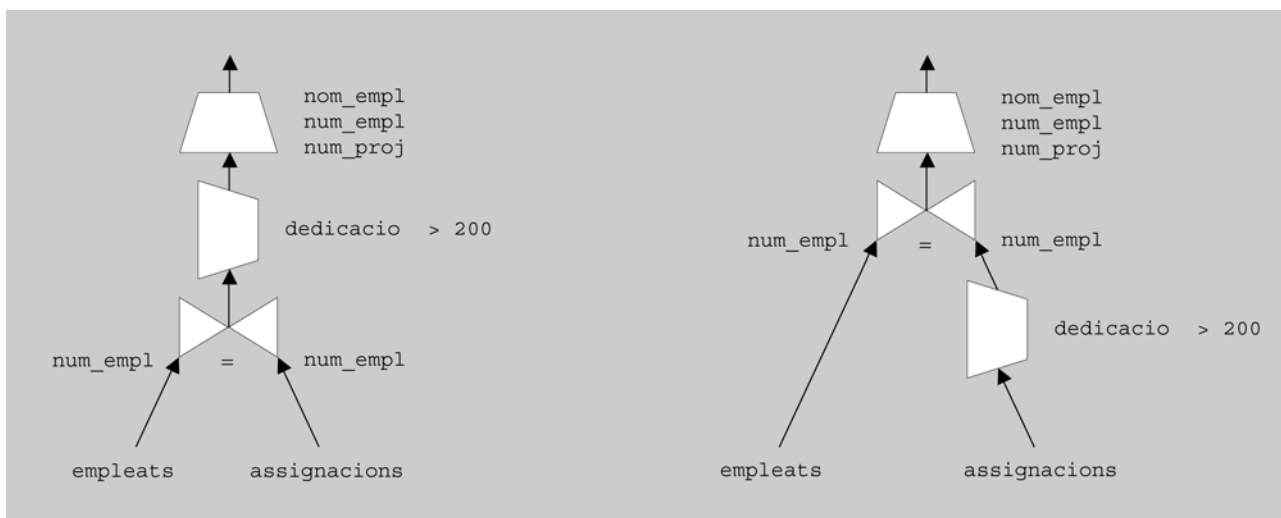
Recordeu que hem estudiat les comandes d'SQL en el mòdul "El llenguatge SQL" de l'assignatura Bases de dades I.

```
SELECT empleats.nom_empl, empleats.num_empl,
       assignacions.num_proj
FROM empleats, assignacions
WHERE empleats.num_empl = assignacions.num_empl
      AND assignacions.dedicacio > 200;
```

Si ens hi fixem, veurem que en aquesta consulta tenim les tres operacions de l'àlgebra relacional següents:

- Combinació natural: entre els atributs `num_empl` de les relacions `empleats` i `assignacions`.
- Selecció: `assignacions.dedicacio > 200`.
- Projecció: de tots els els atributs només ens interessa `empleats.nom_empl`, `empleats.num_empl` i `assignacions.num_proj`.

Si combinem aquestes operacions podem obtenir, per exemple, els arbres sintàctics que presentem tot seguit:



Com podem veure, tots dos arbres sintàctics són equivalents a la consulta original formulada en SQL. La diferència que hi ha entre tots dos és determinada

per l'ordre en el qual s'executen les operacions de l'àlgebra relacional. En el cas de l'arbre de l'esquerra, primerament s'executa l'operació de combinació natural, mentre que en l'arbre de la dreta en primer lloc es resol l'operació de selecció.

També podeu observar que les fulles de tots dos arbres són les relacions de base implicades en la consulta; que els nodes intermedis són les operacions de l'àlgebra relacional especificades a la consulta, i que l'arrel de tots dos arbres representa la resposta a la consulta.

1.2. El procés d'optimització de consultes


Acabem d'explicar en què consisteix el processament de consultes i el paper que hi té l'àlgebra relacional. En aquest subapartat ens centrarem en l'estudi del procés d'optimització.

L'**objectiu de l'optimització de consultes** és trobar el pla d'accés de la consulta, és a dir, l'estratègia d'execució de la consulta que té associat un cost mínim. Per *cost mínim* entenem que el temps necessari per a trobar la resposta a la consulta formulada per l'usuari ha de ser tan petit com sigui possible.

Hi ha diferents **factors de cost** que influeixen en el temps necessari per a resoldre una consulta, el més important dels quals és el nombre d'accessos que s'han de fer al disc, ja sigui per a llegir o per a escriure dades en relacions intermèdies o en les relacions de base. Al seu torn, el nombre d'accessos depèn bàsicament de les cardinalitats d'aquestes relacions i de la longitud dels seus tuples.

Altres factors de cost...

... serien el temps necessari per a accedir a la memòria principal i el temps d'unitat de procés necessari per a executar les operacions demanades a la consulta.

A continuació justifiquem la importància del fet d'optimitzar consultes. Primerament avaluem el cost aproximat (en termes de nombre de tuples als quals s'ha accedit) de tres estratègies d'execució possibles per a la consulta presentada a l'exemple del subapartat anterior. 

Suposem que volem trobar l'estratègia d'execució òptima de la consulta que ja coneixem:

```
SELECT empleats.nom_empl, empleats.num_empl,
       assignacions.num_proj
FROM empleats, assignacions
```

```
WHERE empleats.num_empl = assignacions.num_empl
      AND assignacions.dedicacio > 200;
```

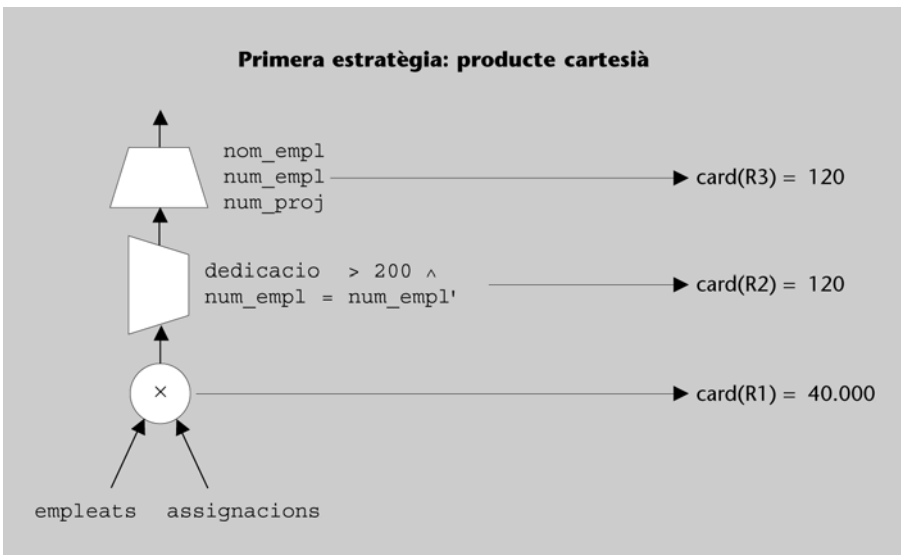
Imaginem que tenim les dades següents: $\text{card}(\text{EMPLEATS}) = 100$, $\text{card}(\text{PROJECTES}) = 20$ i $\text{card}(\text{ASSIGNACIONS}) = 400$, i que un 30% dels empleats que participen en projectes hi dediquen més de 200 hores.

Cardinalitat

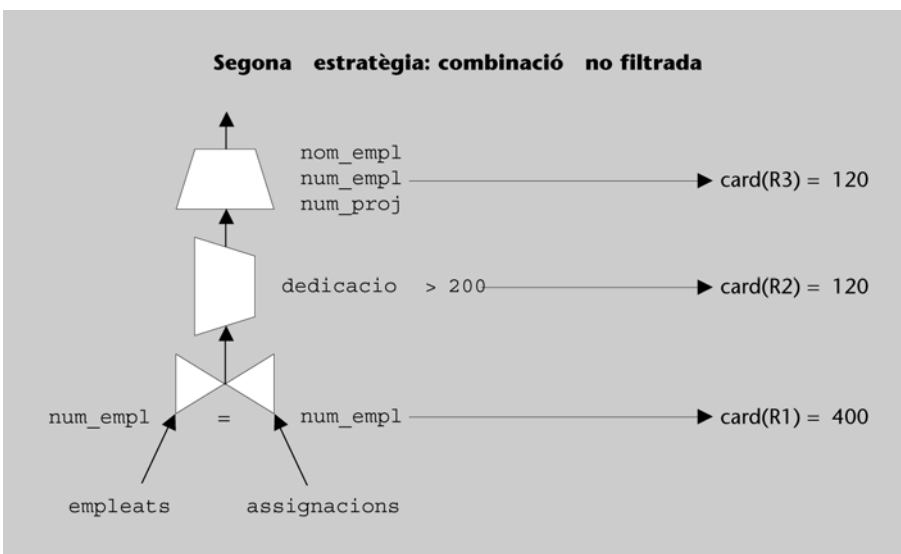
Recordeu que la cardinalitat d'una relació R qualsevol es presenta com $\text{card}(R)$ i és el nombre de tuples que pertanyen a l'extensió de R.

Podem implementar, entre altres, les tres estratègies següents:

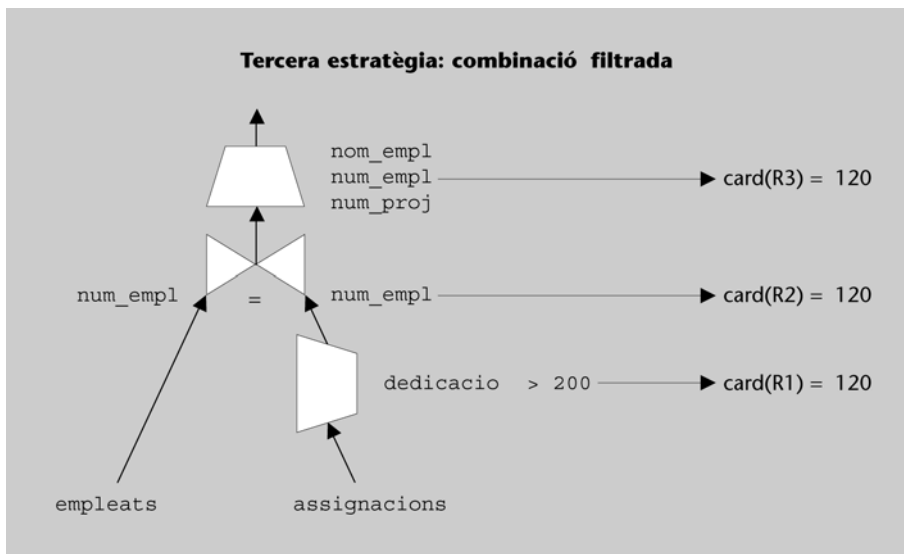
1) Producte cartesià



2) Combinació no filtrada



3) Combinació filtrada




Podem observar que cada estratègia té un cost diferent. La millor estratègia és la combinació filtrada, perquè és la que genera relacions intermèdies de cardinalitat menor i, per tant, serà l'estratègia que implicarà un nombre més reduït d'accessos al disc. Aquesta estratègia executa les instruccions com veiem a continuació:


a) Primerament s'executa l'operació de selecció; aquesta operació dona com a resultat una relació intermèdia amb 120 tuples, ja que sabem que un 30% dels empleats que participen en projectes hi dediquen més de 200 hores.

b) A continuació s'executa l'operació de combinació natural, que genera una relació intermèdia amb 120 tuples, ja que com que l'atribut `num_empl` de la relació `assignacions` és la clau forana que referencia la relació `empleats`, tot tuple de la relació intermèdia anterior trobarà parella a la relació `empleats`.

c) Finalment s'executa l'operació de projecció, que proporciona la resposta a la consulta original formulada per l'usuari; atès que l'atribut `num_empl` és la clau primària de la relació `empleats`, no tindrem dos tuples iguals i, per tant, la resposta estarà composta per 120 tuples.

Acabem de veure que l'ordre d'execució de les operacions formulades dins una consulta repercuteix directament en el cost d'execució de la consulta. Per tant, és important trobar el millor ordre d'execució possible per a les operacions de cada consulta. Malauradament, trobar aquest ordre d'execució requereix molt temps si les consultes no són simples; per això els SGBD relacionals del mercat apliquen mètodes heurístics per a trobar una estratègia d'execució raonablement òptima per a resoldre una consulta determinada. 

L'optimització sintàctica d'una consulta és el procés que determina un ordre d'execució raonablement òptim de les operacions que inclou una consulta.

L'optimització sintàctica de la consulta comença amb la traducció de la consulta original a un arbre sintàctic equivalent i finalitza quan s'ha trobat l'arbre sintàctic òptim. Per a trobar l'arbre sintàctic òptim, l'SGBD relacional aplica els mètodes heurístics següents: 

- 1) Executar les operacions de l'àlgebra relacional que decrements la cardinalitat dels resultats intermedis al més aviat possible.
- 2) Endarrerir al màxim l'execució de les operacions que incrementen la cardinalitat dels resultats intermedis.

Si llegim detingudament les condicions anteriors, veiem que l'optimització sintàctica simplement reordena les operacions de l'àlgebra relacional que figuren a la consulta. En canvi, al nostre exemple hem estimat la cardinalitat de les relacions intermèdies. De fet, l'avaluació del cost de l'arbre sintàctic òptim obtingut com a resultat de l'optimització sintàctica es fa posteriorment, en l'etapa d'optimització física de la consulta.

L'objectiu de l'optimització física d'una consulta és avaluar el cost total de l'execució de l'arbre sintàctic òptim associat a una consulta d'un usuari. Aquest cost és la suma dels costos de totes les operacions que figuren a l'arbre.

El cost de cada operació inclou tant el cost d'execució de l'operació en si com el d'escriure el resultat de l'operació en una relació intermèdia. Per a poder aproximar el cost de les diferents operacions, l'SGBD relacional necessita conèixer la informació següent:

- 1) Estadístiques de la BD, que serveixen per a estimar la longitud dels resultats intermedis i la longitud de les relacions implicades en l'execució de les diferents operacions. Les estadístiques s'emmagatzemen al catàleg de la BD i és missió de l'administrador de la BD mantenir-les actualitzades.

- 2) Les estructures d'emmagatzematge definides a l'esquema intern. D'aquestes estructures, és important tenir informació sobre els aspectes següents:

- a) A quins fitxers s'han emmagatzemat les relacions de base i on es troben aquests fitxers.

Decrement de la cardinalitat

Les operacions de l'àlgebra relacional que decrements la cardinalitat de la relació resultant són la selecció, la projecció, la diferència i la intersecció.

Increment de la cardinalitat

Les operacions de l'àlgebra relacional que incrementen la cardinalitat de la relació resultant són el producte cartesià, la unió i la combinació.

Dades estadístiques

La cardinalitat d'una relació, la longitud dels túbels d'una relació i el nombre de valors diferents que prenen els atributs de les relacions, són exemples de dades estadístiques.

b) Si s'han definit estructures d'agrupació* per als tuples d'una relació segons el valor d'un o més atributs.

c) Si hi ha altres camins d'accés a les dades com, per exemple, índexs secundaris sobre atributs que no són part dels criteris d'agrupació.

3) Els algoritmes d'implementació de les operacions de l'àlgebra relacional disponibles.

Aquestes dades permeten que l'SGBD relacional pugui decidir quins algoritmes són aplicables i quins no. L'aplicabilitat d'aquests algoritmes depèn dels aspectes físics exposats al punt 2).

Per acabar, l'SGBD haurà d'avaluar el cost de cadascun dels algoritmes aplicables i triar l'algoritme que tingui el cost més baix. Per a fer-ho utilitza dades estadístiques.

Quin algoritme cal triar?

Per a l'arbre sintàctic òptim trobat a l'exemple anterior, l'SGBD necessita avaluar el cost de les operacions de combinació natural i de selecció que hi figuren. Suposem que sobre l'atribut `dedicacio` de la relació `assignacions` s'ha definit un índex.

Llavors, per a implementar l'operació de selecció física l'SGBD podria triar entre dos algoritmes:

1) Un algoritme que no fes ús de l'índex. En aquest cas faríem un recorregut seqüencial del fitxer que emmagatzema la relació `assignacions` i comprovaríem la condició demanada a la consulta (`dedicacio > 200`). Per cada tuple que compleixi la condició, projectaríem els atributs que necessitessim (`num_empl` i `num_proj`) i els guardariem en una relació intermèdia R1.

2) Un algoritme que fes ús de l'índex. Aleshores, mitjançant l'índex accediríem només als tuples que verifiquessin la condició, projectaríem els atributs que ens interessessin i guardariem el resultat en una relació intermèdia R1.

Dels dos algoritmes, intuïtivament veiem que el segon segurament tindrà associat un cost més baix, i seria l'algoritme que triaria l'SGBD.

Hem estudiat els dos tipus d'optimització que efectuen els SGBD relacionals del mercat. Per a finalitzar l'estudi de la primera part d'aquest mòdul veurem un tercer tipus d'optimització possible des d'un punt de vista teòric: l'optimització semàntica.

L'objectiu de l'optimització semàntica d'una consulta és simplificar les consultes formulades pels usuaris mitjançant l'ús de la informació que proporcionen les regles d'integritat que s'hagin definit sobre la BD i les lleis de la lògica.

No analitzarem detalladament l'optimització semàntica d'una consulta; només la il·lustrarem amb un exemple.

* Les estructures d'agrupació són les estructures que en anglès s'anomenen *cluster*.

Els algoritmes...


... no implementen una única operació de l'àlgebra relacional, sinó que aquestes operacions són agrupades dins una mateixa operació física. Per exemple, els algoritmes de selecció implementen com una sola operació física l'operació de selecció i opcionalment l'operació de projecció.

Intel·ligència de l'optimitzador semàntic

Suposem, a partir de les relacions de la BD que hem fet servir als exemples anteriors, que un usuari formula la consulta següent:

```
SELECT E2.nom_empl  
FROM empleats E1, empleats E2  
WHERE E1.num_empl = E2.cap AND E2.salari > E1.salari;
```

Aquesta consulta troba el nom dels empleats que guanyen més que el seu superior. Suposem ara que tenim una regla d'integritat a l'esquema de la BD que ens diu que cap empleat no pot guanyar més que el seu superior. En aquest cas, si l'SGBD disposés d'un optimitzador semàntic podria saber que no necessita executar aquesta consulta perquè el resultat serà buit. Per tant, no caldria que l'SGBD fes l'optimització sintàctica ni l'optimització física de la consulta.

L'optimització semàntica de consultes és un tema de recerca de gran interès per a la comunitat científica dedicada a l'estudi de les BD, i és possible que es pugui incorporar als SGBD comercials en un futur pròxim. 


2. Eines d'administració

La segona part d'aquest mòdul didàctic està dedicada a presentar les eines que habitualment incorporen els SGBD del mercat per a facilitar les tasques d'administració i control de les BD.

Com ja sabeu, és necessari que es portin a terme tot un seguit de funcions centralitzades de gestió i administració per a assegurar que l'explotació de les BD és correcta. La persona responsable d'aquestes funcions d'administració és **l'administrador de la base de dades (ABD)**. Per a facilitar la feina de l'ABD, els SGBD incorporen un conjunt d'eines conegudes com **eines d'administració**. Actualment, la majoria d'aquestes eines estan dotades d'interfícies visuals amb finestres, menús desplegable i icones, de manera que l'ABD pot navegar-hi fent ús del ratolí de l'ordinador.

En general, l'administrador de la BD no és un únic usuari, sinó un grup d'usuaris.

El programari que incorpora els SGBD per a facilitar les tasques de gestió i administració de l'ABD és *l'eina d'administració*.


A continuació veurem les eines de suport que ofereixen els diferents SGBD per a ajudar l'ABD a desenvolupar les seves funcions; com que teniu a la vostra disposició l'Informix Personal, farem referència a les eines que incorpora aquest producte*. De totes maneres, tots els productes comercials proporcionen eines molt similars, i per tant, si teniu oportunitat de conèixer o treballar amb altres SGBD, els coneixements que assolieu en aquest mòdul didàctic us seran útils. 

* Quan parlem d'Informix, ens referim a les eines que incorpora la versió 7.22.TC2.

També és important que tingueu en compte que algunes de les eines que presentarem no són d'ús exclusiu de l'ABD, sinó que també estan a l'abast d'altres usuaris, com per exemple el dissenyador de la BD i els mateixos usuaris de les BD.

Cal destacar que qualsevol de les funcionalitats que s'ofereixen mitjançant interfícies visuals també poden obtenir-se mitjançant l'execució de sentències SQL.

2.1. Eines d'administració central

En general el conjunt d'eines d'administració disponibles poden ser cridades des d'una eina d'administració central, o es pot accedir a cadascuna per separat. En el cas particular de l'Informix Personal aquesta eina d'administració central és el *Command Center*, i ofereix els serveis següents: 

1) **Operacions relacionades amb el servidor de la BD** entre les quals podem destacar les següents:

a) La posada en marxa i la desconnexió del servidor de la BD.

b) L'especificació de la modalitat d'operació o de treball del servidor.

2) **Crida a altres eines d'administració**, de les quals les que s'utilitzen més sovint són les tres següents:

a) Eines de gestió d'espais: serveixen per a crear l'espai d'emmagatzematge per a les diferents BD i faciliten la gestió posterior d'aquest espai.

b) Eines de gestió d'esquemes: s'utilitzen per a facilitar la creació dels esquemes de les BD i ajudar al seu manteniment i evolució posterior. També es poden fer servir aquestes eines per a explorar de manera ràpida els esquemes de les BD existents.

c) Eines de recuperació: estan destinades a garantir la recuperabilitat de les dades en casos d'avaries i/o destrucció de la BD. Serveixen per a fer les còpies de seguretat de les dades (bolcats) i dels diferents dietaris de les BD i per a gestionar-les.

3) **Supervisió del funcionament del servidor**. Aquest servei proporciona informació sobre els aspectes següents:

a) Les sessions de treball actives dels diferents usuaris.

b) L'estat en què es troben les activitats destinades a garantir la recuperabilitat de les dades.

c) Possibilitat de controlar les situacions d'emergència. Dins aquest servei podem diferenciar els dos aspectes següents:

- Control de l'estat dels espais d'emmagatzematge i dels camins d'accés a les dades.
- Avís i ajuda per a resoldre situacions d'emergència.

d) Dispositius per a generar informes d'incidències.

Exemple d'una situació d'emergència

Imaginem que l'ABD no ha dimensionat correctament l'espai necessari per a emmagatzemar una determinada BD i que l'espai lliure disponible està per sota d'un cert nivell de seguretat especificat per l'ABD; aquesta situació generaria l'avís d'una situació d'emergència que l'ABD hauria de solucionar afegint, per exemple, més espai d'emmagatzematge.

A continuació, analitzarem més detalladament les eines de gestió d'espais i les eines de gestió d'esquemes. En el cas concret d'Informix Personal, aquestes eines són, respectivament, l'*Space Explorer* i el *Database Explorer*, i poden ser cridades des del *Command Center* o directament.

Modalitats d'operació

Dos exemples de modalitats d'operació són els següents:

- La modalitat en línia (*on line*), que permet que els usuaris autoritzats puguin treballar amb l'SGBD.
- La modalitat administració (*administration*), que permet que l'ABD pugui fer tasques d'administració de manera aïllada; és a dir, sense la presència d'usuaris.

1) *Space Explorer*

L'*Space Explorer* només pot ser utilitzat per l'ABD i proporciona els serveis següents:

a) Visualització dels espais d'emmagatzematge que s'han creat per a guardar les diferents BD i els objectes grans*. Bàsicament, per a cada espai existent proporciona dades respecte al seu nom, la localització i la mida.

* En anglès els objectes llargs s'anomenen *large objects* o, simplement, LOB.

b) Creació de nous espais d'emmagatzematge de BD i espais per a emmagatzemar els objectes grans. Per a cada espai que l'ABD vol crear haurà de proporcionar el nom, la localització, la mida de les pàgines i la mida total del nou espai d'emmagatzematge.

Vegeu els components d'emmagatzematge més rellevants al mòdul didàctic "Components d'emmagatzematge d'una base de dades" d'aquesta assignatura.

c) Modificació dels espais d'emmagatzematge existents. Aquestes modificacions estan destinades a afegir capacitat d'emmagatzematge als diferents espais.

d) Esborrament dels espais d'emmagatzematge.

Adicionalment, en els SGBD grans (no en edicions personals com ara l'Informix Personal), des d'aquest tipus d'eina és possible crear espais de mirall*.

Aquests espais són una segona còpia d'una BD (que seria la còpia primària) i tenen com a finalitat garantir la màxima disponibilitat de les dades. Això implica que, en el cas que hi hagi algun problema amb la còpia primària, els usuaris passarien a treballar amb la segona còpia mentre l'SGBD fa les tasques destinades a recuperar la còpia primària.

* En anglès els espais de mirall s'anomenen *mirroring*.

L'existència d'espais de mirall...

... complica la gestió de les transaccions, atès que l'SGBD haurà de mantenir la concordança entre la còpia primària i la còpia secundària.

2) *Database Explorer*

Un cop que s'han creat els espais destinats a emmagatzemar les BD, els dissenyadors i altres usuaris autoritzats per l'ABD poden crear els esquemes de les BD. També poden visualitzar els esquemes de les BD existents i modificar los. Per a facilitar aquestes tasques, els SGBD proporcionen eines de gestió d'esquemes. En el cas de l'Informix Personal, aquesta eina és el *Database Explorer*, que incorpora els serveis següents:

a) Visualització dels esquemes de les BD existents. Per a cada BD és possible consultar-ne l'esquema fent clic successivament sobre les diferents icones que apareixen. Bàsicament, els usuaris autoritzats poden trobar la informació següent:

- La manera com ha estat definida cadascuna de les taules. En concret, l'eina proporciona aquestes dades:
 - Les columnes de la taula: nom, tipus, precisió i escala (només per tipus numèrics); valors per defecte, indicació de si admet valors nuls i referència de quines columnes són la clau primària.

- Els índexs que s’han definit sobre les columnes: nom i tipus d’índex.
- Les claus foranes: quines columnes són clau forana i quines taules referencien.
- Altres propietats de les taules com, per exemple, el propietari de la taula, la data de la darrera modificació, la longitud en bytes de cada fila, les mides de les extensions* de la taula, el nombre actual de files que té la taula, el nivell de granularitat de la taula per a la gestió de transaccions, etc.
- Les vistes que s’hagin definit sobre les diferents taules.
- Els procediments emmagatzemats.
- Els disparadors*.

* En anglès *extents*.

* En anglès els disparadors s’anomenen *triggers*.

b) Creació de l’esquema d’una nova BD. Per a crear BD noves, els dissenyadors hauran d’especificar el nom de la nova BD i quin serà l’espai d’emmagatzematge d’aquesta BD. També caldrà que especifiquin com volen que es gestionin els dietaris de la nova BD. Un cop finalitzades aquestes operacions, els dissenyadors hauran de definir les taules de la nova BD, les vistes de la nova BD, els procediments emmagatzemats i els disparadors.


Els dietaris d’una BD...

... són els fitxers on s’emmagatzema part de la feina que les transaccions han efectuat en la BD. La seva funció és garantir les propietats d’atomicitat i definitivitat de les transaccions.

c) Modificació de l’esquema de les BD. Aquest servei està destinat a facilitar les tasques relacionades amb l’evolució dels esquemes. Bàsicament hauria de permetre afegir, esborrar i modificar les taules, les vistes, els procediments emmagatzemats i els disparadors de cadascun dels esquemes de les BD existents.

En el cas concret de l’Informix Personal i des del *Database Explorer*, no és possible consultar, crear i modificar directament la definició de les vistes, dels procediments emmagatzemats i dels disparadors. Per a poder fer aquestes tasques, caldrà utilitzar les sentències que incorpora SQL. Per a consultar la definició dels elements anteriors, el dissenyador haurà de fer consultes al catàleg de la BD, i per a crear nous elements o modificar-los haurà d’utilitzar també SQL.

2.2. Eines d’edició i manipulació de les BD

Per a facilitar les tasques de manipulació de les BD, utilitzant directament SQL, els SGBD del mercat incorporen programari addicional que podríem classificar com eines d’edició. En l’Informix Personal, aquest programari és **SQL Editor**, i de fet és la darrera eina que veurem dins aquest mòdul didàctic. 

L’SQL Editor és a l’abast de tots els usuaris autoritzats a treballar amb l’SGBD i, un cop s’ha seleccionat una BD, permet fer les operacions següents:

1) Crear, recuperar i modificar documents d’SQL. Aquests documents són fitxers de text* en els quals els usuaris poden guardar les seves comandes d’SQL.

* Sovint aquests fitxers de text s’anomenen *scripts*.

2) Introduir noves comandes d'SQL. Les comandes s'introdueixen en una pàgina d'edició. També és possible fer comprovacions sintàctiques de les comandes d'SQL introduïdes. Finalment, es pot guardar aquestes sentències dins un document d'SQL.

3) Executar comandes d'SQL. Els usuaris poden seleccionar les sentències d'SQL que volen executar. Els resultats de les comandes de consulta surten en pàgines addicionals. Els usuaris també poden emmagatzemar en fitxers de text el resultat d'aquestes consultes.

L'ABD pot utilitzar l'SQL Editor...

... per exemple, per a actualitzar les estadístiques de les BD (comanda UPDATE STATISTICS de l'SQL) i per a concedir i revocar permisos (comandes GRANT i REVOKE de l'SQL).

2.3. Resum de les eines d'administració

Hem estudiat les eines d'administració que incorpora l'Informix Personal per a facilitar la tasca de l'ABD i d'altres usuaris. En síntesi podem classificar aquest conjunt d'eines en dos grans grups:

- 1) Eines d'administració centrals: aquest grup inclou eines de gestió d'esquemes, de gestió d'espais, de recuperació, etc.
- 2) Eines d'edició i manipulació de les BD.

La taula següent mostra les eines d'administració que incorporen altres SGBD que hi ha al mercat; aquestes eines ofereixen serveis similars als que acabem de veure i per això utilitzem la classificació anterior:

| SGBD comercials | Eines centrals d'administració | Eines d'edició (editors) |
|----------------------------------|--|-----------------------------------|
| Informix (v7, v9 i Personal) | Command Center (IECC) Schema Knowledge Informix Server Administrator (ISA) | SQL Editor DB Access |
| DB2 d'IBM (v5) | Control Center | Command Center |
| Oracle (v7, v8 i Personal) | Schema (Security, Storage...) Navigator | SQL Plus WorkSheet |
| SQL Server Microsoft (v6.5 i v7) | SQL Enterprise Manager | ISQL/w MS Query Query Analyzer |

Resum

En aquest mòdul didàctic hem tractat de dos temes relacionats amb les BD clarament diferenciats: l'optimització de consultes i les eines d'administració.

1) En el primer apartat hem estudiat l'**optimització de consultes**, hem vist en què consisteix aquest procés i com influeix en el rendiment global de l'SGBD. L'objectiu de l'optimització de consultes és trobar el pla d'accés d'una consulta. Per a trobar aquest pla d'accés, l'SGBD porta a terme tres tipus d'optimització de consultes: la sintàctica, la física i la semàntica. Cal recordar, però, que l'optimització semàntica no es troba implementada en els productes comercials.

2) En el segon apartat hem vist les **eines d'administració** que normalment incorporen els SGBD del mercat per a facilitar la feina dels ABD i dels diferents usuaris de les BD. Bàsicament hem classificat aquestes eines en dues grans famílies:

- Eines d'administració centrals que inclouen, entre d'altres, les eines de gestió d'esquemes, les de gestió d'espais i les de recuperació.
- Eines d'edició i manipulació de les BD.

Activitats

1. Exploreu les diferents eines d'administració de l'Informix Personal estudiades en aquest mòdul i intenteu trobar altres eines que incorpora aquest programari i que no heu estudiat. Esbrineu quins són els serveis que proporcionen aquestes eines.

Exercicis d'autoavaluació

1. Imaginem una BD que contingui, entre d'altres, les relacions següents:

- EMPLEATS(num_empl, nom_empl, num_dept, categoria, sou).
- DEPARTAMENTS(num_dept, nom_dept, director, ciutat_dept).
- CATEGORIES_LABORALS(categoria, sou_màxim).

Els atributs subratllats són la clau primària en cada relació. A més:

- L'atribut *categoria* de la relació *EMPLEATS* és una clau forana que referencia la relació *CATEGORIES_LABORALS*.
- L'atribut *num_dept* de la relació *EMPLEATS* és una clau forana que referencia la relació *DEPARTAMENTS*.
- L'atribut *director* de la relació *DEPARTAMENTS* és una clau forana que referencia la relació *EMPLEATS*.

Suposem que un usuari està interessat a saber quins són els empleats que treballen en departaments situats a la ciutat de Barcelona i que tenen un sou per sota del 90% del sou màxim assignat a la seva categoria laboral. Concretament, l'usuari vol saber-ne el nom, el sou, la categoria laboral, el sou màxim assignat a aquesta categoria i el nom del departament on treballen aquests empleats.

- Escriuiu una solució en SQL per a la consulta d'aquest usuari.
- Trobeu un arbre sintàctic equivalent a la consulta d'SQL anterior.

2. Tenim una BD que conté, entre d'altres, les relacions següents:

- PROVEÏDORS(codi_prov, nom_prov, carrer, número, ciutat, telèfon).
- MATERIALS(codi_mat, nom_mat, preu_unitat).
- SUBMINISTRAMENTS(codi_prov, codi_mat, quant_total).

Els atributs subratllats són la clau primària dins de cada relació. A més:

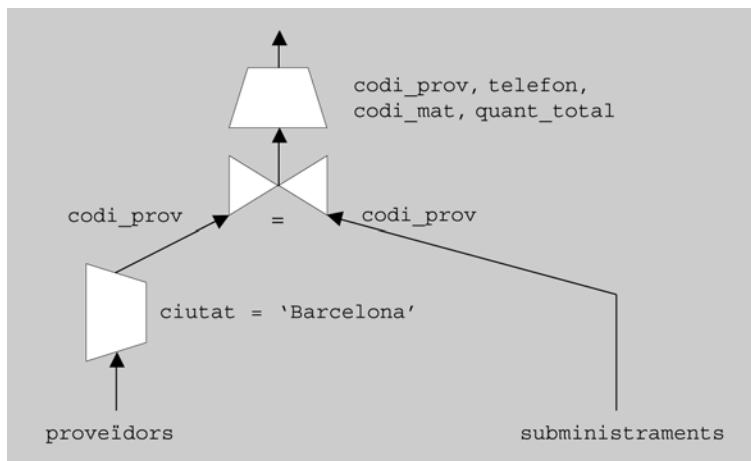
- L'atribut *codi_prov* de la relació *SUBMINISTRAMENTS* és una clau forana que referencia la relació *PROVEÏDORS*.
- L'atribut *codi_mat* de la relació *SUBMINISTRAMENTS* és una clau forana que referencia la relació *MATERIALS*.

Sabem, a més, que la cardinalitat de les diferents relacions és: $\text{card}(\text{PROVEÏDORS}) = 100$, $\text{card}(\text{MATERIALS}) = 1.000$ i $\text{card}(\text{SUBMINISTRAMENTS}) = 10.000$, i que un 5% dels proveïdors subministren el material amb codi m2.

Suposem que un usuari vol saber quins són els proveïdors que subministren el material amb codi igual a m2. Concretament, l'usuari vol trobar el codi d'aquests proveïdors, els seus telèfons i la quantitat total que poden subministrar.

- Escriuiu una solució en SQL per a la consulta d'aquest usuari.
- Trobeu l'arbre sintàctic òptim equivalent a la consulta anterior obtingut com a resultat de baixar les operacions de selecció tan avall com sigui possible en l'arbre.
- Estimeu les cardinalitats de les relacions intermèdies de l'arbre sintàctic òptim obtingut.

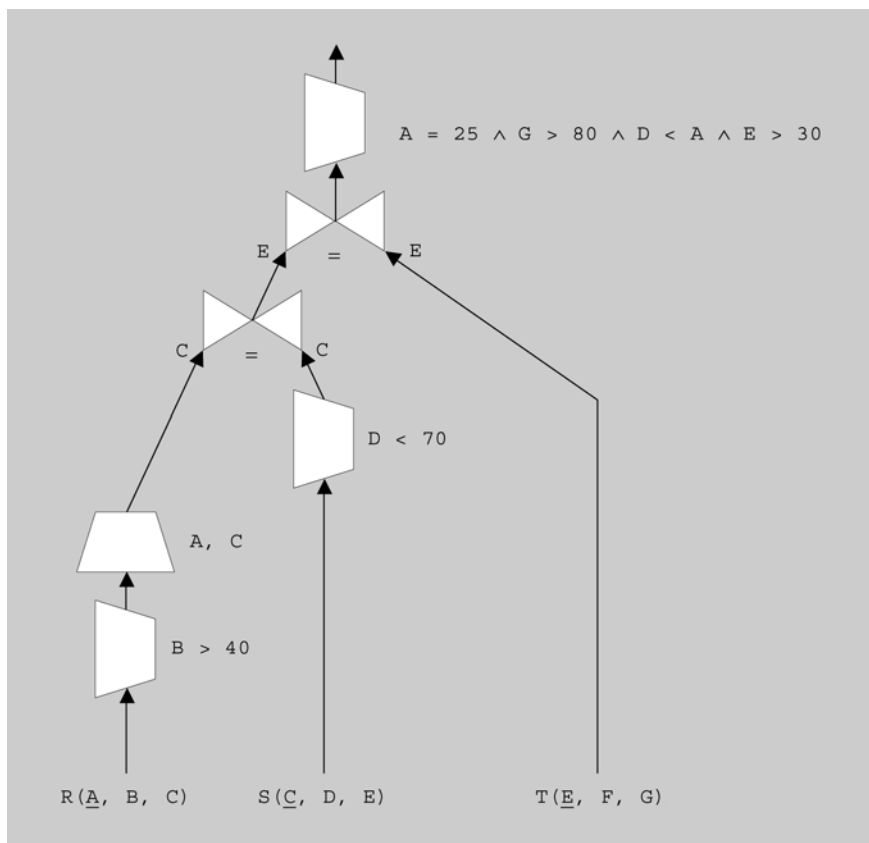
3. Suposem que tenim l'arbre sintàctic següent:



Aquest arbre representa una consulta formulada sobre les relacions de la BD de l'exercici d'autoavaluació 2.

- a) Escriviu una consulta en SQL que pugui donar com a resultat l'arbre sintàctic anterior.
- b) Tenint en compte les cardinalitats de les relacions de la BD donades a l'exercici anterior, raoneu si la cardinalitat de la relació intermèdia R1 que resulta d'haver executat l'operació de combinació natural serà igual, més petita o més petita o igual a 10.000 tuples. Raoneu breument la resposta.

4. Donat l'arbre sintàctic següent (els atributs subratllats en el gràfic són les claus primàries dins de cada relació):



- a) Baixeu les operacions de selecció tan avall com sigui possible en l'arbre anterior amb l'objectiu de trobar l'arbre sintàctic òptim.
- b) És possible fer alguna optimització semàntica sobre l'arbre original utilitzant les lleis de la lògica?

Solucionari

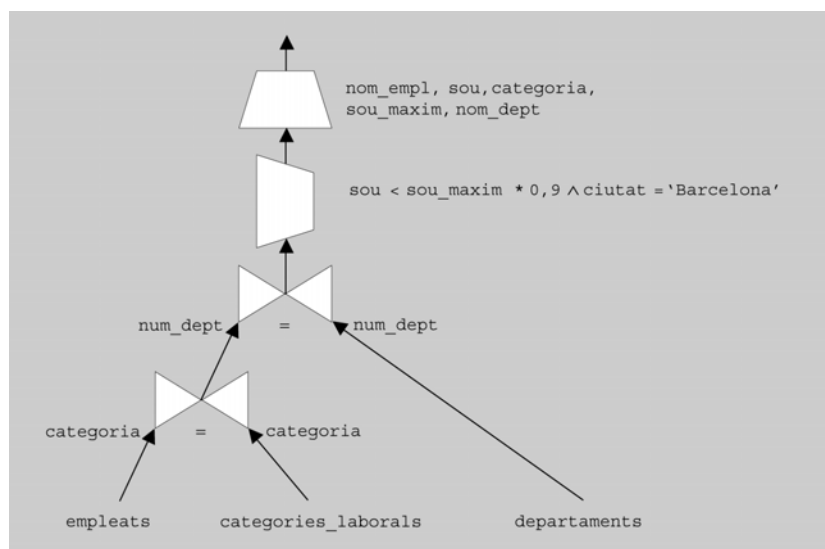
Exercicis d'autoavaluació

1.

a) L'expressió en llenguatge SQL de la consulta és la següent:

```
SELECT e.nom_empl, e.sou, e.categoria, c.sou_maxim, d.nom_dept
FROM empleats e, categories_laborals c, departaments d
WHERE e.num_dept = d.num_dept AND e.categoria = c.categoria
      AND e.sou < c.sou_maxim * 0.9 AND d.ciutat = 'Barcelona';
```

b) Podem trobar diferents arbres sintàctics equivalents a la consulta anterior. Un dels arbres sintàctics correctes és el següent:



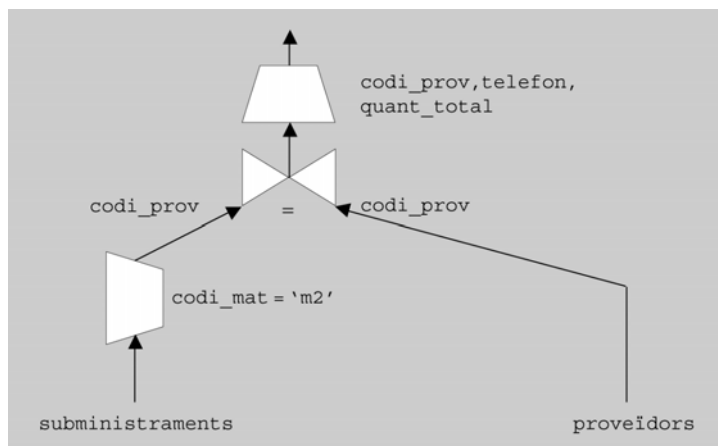
Aquest arbre reflecteix totes les operacions de l'àlgebra relacional incloses a la consulta d'SQL: dues operacions de combinació natural, dues operacions de selecció (que es poden agrupar dins una mateixa operació de selecció) i una operació de projecció.

2.

a) L'expressió en SQL de la consulta és la següent:

```
SELECT p.codi_prov, p.telefon, s.quant_total
FROM proveidors p, subministraments s
WHERE p.codi_prov = s.codi_prov AND s.codi_mat = 'm2';
```

b) L'arbre sintàctic òptim associat a la consulta anterior és el següent:



Aquest arbre recull les tres operacions de l'àlgebra relacional que intervenen en la resolució de la consulta anterior: una operació de projecció, una de combinació natural i una de selecció; és l'arbre sintàctic òptim (tenint en compte les especificacions de l'enunciat), atès que prèviament a l'operació de combinació natural hem filtrat els tuples que compleixen la condició especificada a l'operació de selecció.

c) Les cardinalitats de les relacions intermèdies que són resultat de l'execució de les operacions de l'àlgebra relacional de l'arbre anterior són les següents:

- $\text{card}(R1) = 10.000 \cdot 0,05 = 500$. R1 és la relació intermèdia resultat d'aplicar l'operació de selecció sobre la relació `subministraments`; com que sabem que un 5% dels proveïdors subministren el material m2 la cardinalitat estimada d'aquesta relació és de 500 tuples.
- $\text{card}(R2) = \text{card}(R1) = 500$. R2 és la relació intermèdia que resulta d'executar l'operació de combinació natural entre la relació intermèdia R1 i la relació `proveïdors`; atès que `subministraments.codi_prov` és una clau forana que referencia la relació `proveïdors` i que aquest atribut està inclòs a R1, de segur que tot tuple d'R1 trobarà parella a la relació `proveïdors` i, per tant, la cardinalitat d'R2 serà igual a 500 tuples.
- $\text{card}(R3) = \text{card}(R2) = 500$. R3 és la relació intermèdia que es genera després de l'execució de l'operació de projecció i, de fet, és el resultat de la consulta; atès que no tenim elements repetits, la cardinalitat d'R3 també serà igual a 500 tuples.

3.

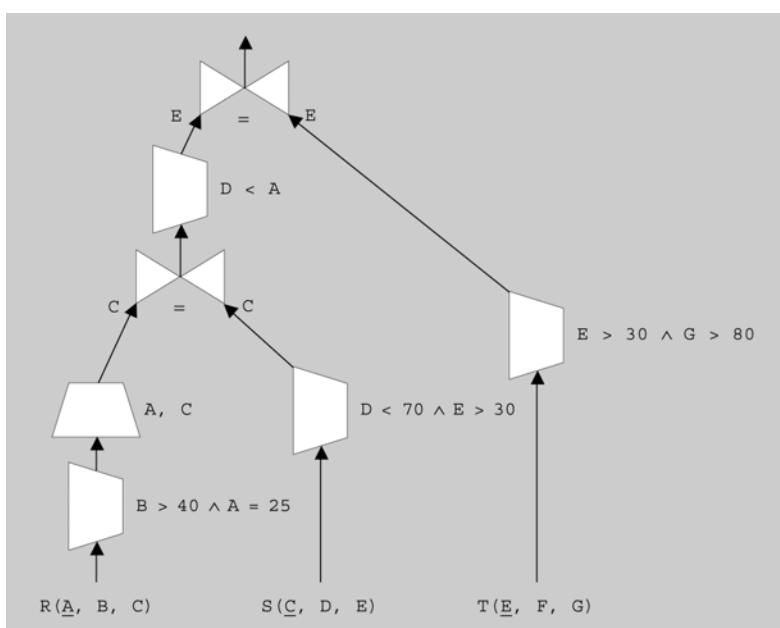
a) L'expressió en SQL que correspon a la figura indicada és la següent:

```
SELECT e.codi_prov, e.telefon, s.codi_mat, s.quant_total
FROM proveïdors p, subministraments s
WHERE p.ciutat = 'Barcelona' AND p.codi_prov = s.codi_prov;
```

b) La cardinalitat de la relació intermèdia R1 que resulta d'haver aplicat l'operació de selecció serà menor o igual a la cardinalitat de la relació `subministraments`, que és de 10.000 tuples; com que hem filtrat tuples de la relació `proveïdors`, ara no podem assegurar que tots els tuples de la relació `subministraments` (recordeu que `subministraments.codi_prov` és una clau forana que referencia la relació de `proveïdors`) trobaran parella a la relació intermèdia que guarda els resultats de l'operació de selecció. Si tots els proveïdors són de la ciutat de Barcelona, llavors l'operació de selecció no filtra cap tuple i, en aquest cas concret, $\text{card}(R1) = 10.000$. De manera anàloga, si els proveïdors de Barcelona no subministren cap producte (imaginem que ens limitem a guardar les dades personals d'aquests proveïdors), la $\text{card}(R1) = 0$. En resum, no tenim dades suficients per a afirmar quin és la cardinalitat d'R1, només podem donar-ne una fita superior i una fita inferior; de manera que $0 \leq \text{card}(R1) \leq 10.000$.

4.

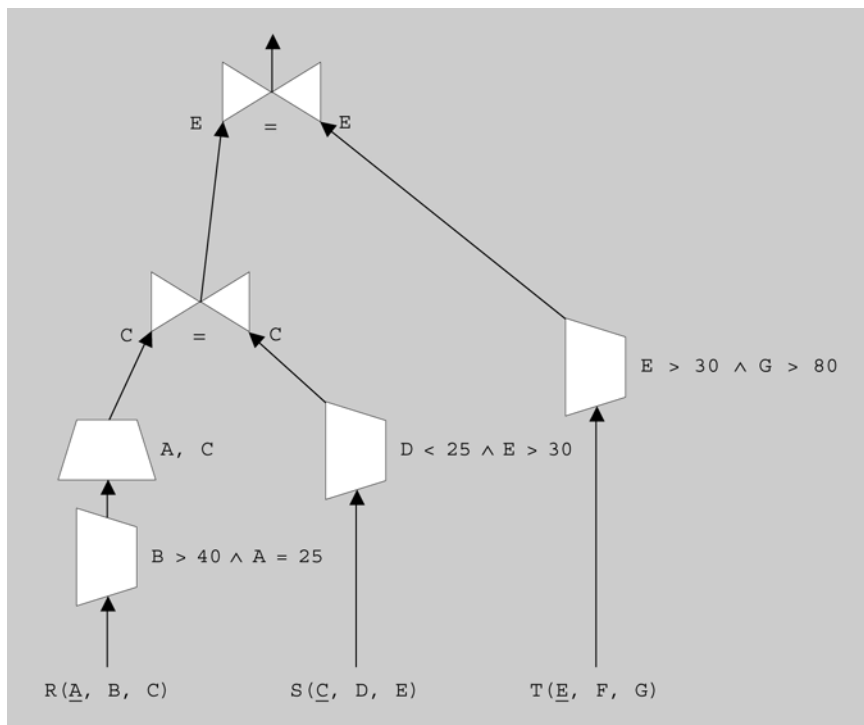
a) L'arbre sintàctic òptim després d'haver baixat les operacions de selecció és el següent:



Aquest és l'arbre sintàctic òptim que tenim després d'haver baixat les operacions de selecció $A = 25 \wedge G > 80 \wedge D < A \wedge E > 30$. Analitzant cadascuna de les restriccions de manera independent trobem el següent:

- La restricció $A = 25$ pot baixar per la branca de la relació R , ja que és un atribut que pertany a aquesta relació i és possible agrupar-la amb la restricció sobre l'atribut B (on $B > 40$) en una única operació de selecció.
- La restricció $G > 80$ baixa per la branca de la relació T , atès que és un atribut que pertany a aquesta relació.
- La restricció $D < A$ només pot baixar per sota de l'operació de combinació natural sobre l'atribut E , perquè els atributs D i A són de relacions diferents (de R i S , respectivament) i fins que no s'executi l'operació de combinació natural sobre l'atribut C , no es pot aplicar l'operació de selecció.
- Per acabar la restricció $E > 30$ baixarà per les branques de les relacions de S i de T , atès que és un atribut que es troba a totes dues relacions; en tots dos casos, la restricció es pot agrupar amb les operacions de selecció existents.

b) Sí que és possible. Si estudiem detalladament les restriccions que apareixen en la darrera operació de selecció, arribem a la conclusió que n'hi ha prou d'assegurar que $D < 25$, ja que el valor de l'atribut A està fixat a 25. Si substituïm les restriccions de l'operació de selecció de l'arbre original per l'expressió equivalent $A = 25 \wedge G > 80 \wedge D < 25 \wedge E > 30$, i a continuació trobem l'arbre sintàctic òptim, arribarem a l'arbre següent:



Si comparem aquest arbre amb l'arbre anterior podem veure que ens hem estalviat l'operació de selecció que comprovava que $D < A$, i que l'operació de selecció de la branca de la relació S també ha variat; en aquest darrer cas, hem tornat a fer una optimització semàntica aplicant les lleis de la lògica; en concret hem simplificat l'expressió $D < 70 \wedge D < 25 \wedge E > 30$ i ha estat substituïda per l'expressió equivalent $D < 25 \wedge E > 30$.

Glossari

ABD: administrador de la base de dades.

arbre sintàctic d'una consulta *m* Representació interna en forma d'arbre d'una consulta en què les fulles de l'arbre representen les relacions de base implicades en la consulta, els nodes intermedis corresponen a les operacions demanades a la consulta i l'arrel de l'arbre és la resposta a la consulta formulada.

BD *f* Base de dades.

eines d'administració *f* Programari que incorporen els SGBD per a facilitar les tasques de gestió i administració de l'ABD o d'altres usuaris de les BD.

factors de cost *m* Aspectes que influeixen en el cost total d'execució d'una consulta.

optimització de consultes *f* Procés que porta a terme l'SGBD destinat a determinar quina és la millor estratègia d'execució d'una consulta.

optimització física d'una consulta *f* Procés que avalua el cost total d'execució d'una consulta i determina quins algorismes s'han d'utilitzar per a resoldre les diferents operacions de la consulta, tenint en compte les característiques de les estructures d'emmagatzematge i dels camins d'accés que s'han definit per a accedir a les dades.

optimització semàntica d'una consulta *f* Procés orientat a simplificar les consultes formulades pels usuaris mitjançant l'ús de la informació que proporcionen les regles d'integritat que s'hagin definit sobre la BD i les lleis de la lògica.

optimització sintàctica d'una consulta *f* Procés que determina quin és l'ordre òptim d'execució de les operacions de l'àlgebra relacional incloses en una consulta.

pla d'accés d'una consulta *m* Estratègia d'execució d'una consulta que té associat un cost mínim.

SGBD *m* Sistema de gestió de bases de dades.

SQL *m* *Structured Query Language*.

Bibliografia

Elmasri, R.; Navathe, S.B. (2000). *Sistemas de bases de datos. Conceptos fundamentales* (3a. ed.). Madrid: Addison-Wesley Iberoamericana.

Gardarin, G.; Valduriez, P. (1989). *Relational Databases and Knowledge Bases*. Reading (Massachusetts): Addison-Wesley.