Disseny multidimensional i explotació de dades

PID_00270635

Alberto Abelló Gamazo
Carles Llorach Rius
Víctor Ruiz Marqués

Temps mínim de dedicació recomanat: 12 hores

Universitat Oberta de Catalunya

Alberto Abelló Gamazo

Doctor i enginyer en Informàtica per la Universitat Politècnica de Catalunya. Professor associat al Departament de Llenguatges i Sistemes Informàtics d'aquesta universitat.

Carles Llorach Rius

Màster en Gestió d'Empreses - MBA per la Universitat Rovira i Virgili i enginyer en Informàtica per la Universitat Politècnica de Catalunya. Professor col·laborador a la Universitat Oberta de Catalunya.

Víctor Ruiz Marqués

Enginyer tècnic en Electrònica Industrial (especialitat en Automàtica) per la Universitat Politècnica de Catalunya. Enginyer tècnic en Informàtica de Gestió i enginyer en Informàtica per la Universitat Oberta de Catalunya. Consultor especialitzat en projectes de Business Intelligence i de ERP/CRM. Professor col·laborador a la Universitat Oberta de Catalunya.

L'encàrrec i la creació d'aquest recurs d'aprenentatge UOC han estat coordinats per la professora: Àngels Rius Gavidia

Primera edició: febrer 2020

Av. Tibidabo, 39-43, 08035 Barcelona
Realització editorial: FUOC

Índex

Introducció

No n'hi ha prou amb tenir un magatzem de dades per disposar de tota la informació que ens fa falta i ser capaços d'utilitzar-la per prendre decisions. La gran quantitat de dades i la falta d'especialització en informàtica per part dels analistes fan que resulti imprescindible disposar d'algun tipus d'eina que en faciliti la consulta.

D'aquesta manera, a partir del magatzem de dades corporatiu, se solen dissenyar petits magatzems de dades departamentals que acosten les dades als usuaris. Aquests magatzems es dissenyen utilitzant el model multidimensional, de manera que es puguin utilitzar eines OLAP per consultar-los. Les eines OLAP i els magatzems de dades no s'exclouen, es complementen.

A grans trets, el model multidimensional distingeix dos tipus de dades: els Fets que volem analitzar i les Dimensions que utilitzem per analitzar-los. Aquesta divisió produeix dos beneficis: d'una banda, podem utilitzar tècniques específiques d'emmagatzematge i accés de dades; i, de l'altra, facilitem la comprensió de les dades de manera que els analistes són capaços de formular les seves consultes gairebé sense cap coneixement informàtic i mitjançant eines gràfiques molt intuïtives.

En aquest mòdul trobareu una definició formal del model multidimensional, així com una guia per fer un bon disseny conceptual, lògic i físic. Per facilitar les consultes multidimensionals com a part del disseny físic, es veuran algunes paraules reservades del llenguatge bàsic estàndard SQL'99.

També s'aborden les interrelacions existents entre el disseny multidimensional i el disseny de magatzems de dades: els punts de coincidència, les diferències i les implicacions que comporta adoptar un determinat enfocament de disseny o un altre.

Finalment, la visualització de dades és una preocupació clau per a les organitzacions i els professionals de la intel·ligència de negoci i analistes, ja que afecta de manera determinant al valor que s'obté de la informació.

Quan les condicions de visualització de dades són les adequades, els usuaris poden interactuar amb les dades de manera molt més efectiva. El resultat és l'enriquiment de l'organització, que gaudeix d'un nivell de coneixement com mai i es troba més ben posicionada a l'hora de prendre decisions i incrementar la productivitat.

Afortunadament, disposem d'una àmplia gamma de recursos: metodologies, eines, guies, etc. que ens facilitaran la publicació de les nostres dades, de manera que la seva visualització sigui la més adequada per al destinatari d'aquesta informació.

Objectius

Aquest mòdul didàctic presenta el model multidimensional i alguns conceptes associats. Posteriorment, presenta l'explotació de dades. Amb aquest mòdul, aconseguireu els objectius següents:

1. Conèixer els components del model multidimensional (estructures de dades, operacions i restriccions d'integritat).

2. Entendre quin és el disseny multidimensional i els problemes de disseny que presenta (en l'àmbit conceptual, així com en el lògic i físic).

3. Ser capaços de dissenyar una base de dades multidimensional.

4. Saber què ens ofereix l'estàndard SQL per facilitar les consultes multidimensionals.

5. Entendre els mecanismes d'emmagatzematge i indexació associats a les eines multidimensionals (MOLAP, ROLAP, etc.).

6. Prendre consciència de la importància de la visualització de dades en condicions òptimes per millorar la presa de decisions, fomentar l'intercanvi d'informació i millorar la comunicació.

7. Entendre la utilitat i idoneïtat dels diferents formats de presentació existents.

8. Conèixer l'ampli conjunt de recursos en línia disponibles i els principis que n'han de regir la utilització.

9. Conèixer les eines de suport i presentació de dades existents i estar capacitats per avaluar-les enfront de les necessitats de l'organització.

10. Identificar els riscos que suposa l'explotació de dades de baixa qualitat i també la seva inadequada visualització.

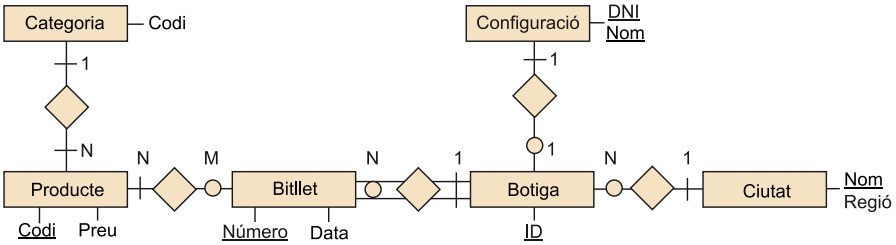
1. Les necessitats dels analistes i les eines OLAP

Fins ara esteu acostumats a veure eines OLTP, que estan basades en la gestió de transaccions (conjunts d'operacions d'alta, baixa, modificació i consulta de dades) que ajuden en el funcionament diari del negoci. La pregunta que us hauríeu de fer és si aquestes eines són adequades per analitzar el funcionament del negoci i prendre decisions. Creieu que un directiu d'una companyia serà capaç de fer una consulta amb SQL sobre un esquema ER o simplement estarà disposat a dedicar el seu valuós temps intentant fer-la?

Nota

El sistema que utilitzarem serà cometes per a les instàncies, cursiva per a termes anglesos i operacions del model multidimensional, tipus de lletra CourierNew per als elements dels esquemes i majúscules per als noms dels elements del model multidimensional i les operacions. A més, en negreta hi haurà la primera aparició dels termes principals que introduïm.

Figura 1



Les dificultats de consulta per als usuaris no informàtics

Un usuari no informàtic ni tan sols entendria un esquema ER tan senzill com el que teniu a la figura 1. A més, no es tracta només de consultar el preu d'un determinat producte, sinó, atesa una cadena de botigues repartides per tot l'Estat, saber com han evolucionat les vendes dels diferents productes durant el mes passat respecte al mateix mes de l'any anterior.

Com ja sabem, el model ER va ser concebut per reduir la quantitat de dades redundants i evitar haver de modificar molts registres alhora en fer un únic canvi. Això s'aconsegueix a costa d'empitjorar el temps de resposta a les consultes (assumint que hi haurà un nombre elevat d'actualitzacions respecte al nombre de consultes). La pregunta que hem de fer-nos ara és la següent: això serveix per als analistes? Què voldran actualitzar aquests usuaris? Res, ja que ells només volen saber com va l'empresa, no introduir-li dades noves.

El model ER no facilita precisament la consulta de les dades. En realitat, probablement pel fet que barreja diferents processos de negoci, als usuaris els resultaria difícil fins i tot recordar l'esquema. A més, tampoc no és fàcil construir un programari que faciliti aquesta consulta.

El model ER, que resulta molt convenient en entorns en els quals es produeixen canvis freqüents, no és gaire adequat per a entorns d'anàlisis en els quals el que preval és tenir respostes ràpides a les consultes i que l'esquema sigui fàcil d'entendre. El sistema ja no ha d'ajudar a vendre, comprar, produir o transportar, sinó a avaluar, comparar, pressupostar, planificar, projectar, etc.

1.1. Bases de dades estadístiques

El primer que se'ns podria ocórrer per solucionar aquest problema és pensar en les bases de dades estadístiques. Aquests sistemes s'utilitzen per a grans estudis socioeconòmics, com l'estudi del cens, la producció nacional o els patrons de consum. Permeten analitzar les dades des de diferents punts de vista i mostrar tot tipus de mesures estadístiques. A la taula següent, podeu veure un exemple del que es podria obtenir amb una d'aquestes eines.

Dades estadístiques

Per acostar-nos al concepte amb més propietat, en lloc de bases de dades estadístiques, hauríem de parlar simplement d'eines que faciliten les consultes estadístiques a bases de dades.

Taula 1

Comptador d'articules venuts		Barcelona	Tarragona	Lleida	Girona
5-1-2000	Bolígrafs	15	3	7	1
	Gomes	3	1	0	5
	Portamines	4	0	6	2

Lectura complementària

Podeu veure exemples de taules estadístiques a l'Anuari estadístic de Catalunya 1992-2001. Col·lecció Estadística de Síntesis. Departament d'Economia i Finances. Institut d'Estadística de Catalunya.

Suma d'ingressos per article		Barcelona	Tarragona	Lleida	Girona
5-1-2000	Bolígrafs	39,5	5,1	15,9	1,2
	Gomes	1,4	0,3	0	1,4
	Portamines	8,7	0	11	5,1

Cal distingir clarament les taules estadístiques de les taules relacionals. En les primeres, podem canviar les files per columnes i viceversa, mentre que en les segones no podem. És el mateix tenir una taula estadística d'articles per població i una de població per articles. No obstant això, cada valor numèric que hi ha a la taula estadística estaria en una fila diferent d'una taula relacional. A més, aquests valors corresponen a atributs acumulats.

Atributs acumulats

Els atributs acumulats s'obtenen com a resultat d'aplicar una funció d'agregació a atributs detallats.

A les bases de dades estadístiques, se'ls demana que garanteixin la confidencialitat de les dades dels individus. Per consegüent, el seu punt fort és la seguretat (de manera més concreta, els mecanismes de protecció d'inferència de dades) i no tant la facilitat de consulta o la presentació dels resultats, com seria

d'esperar si els volguéssim utilitzar per a prendre decisions en una empresa. Se sol tractar simplement de sistemes relacionals amb un programari afegit per millorar la seguretat i les consultes.

Un sistema relacional ja ofereix una certa flexibilitat en l'estructuració i consulta de les dades. No obstant això, obtenir la suma acumulada de vendes combinant totals i subtotals, o determinar un cert rànquing (proporcionar els deu països amb un total més gran de vendes), és molt difícil, si no impossible, amb SQL. En el millor dels casos, seria necessària la intervenció d'un informàtic per fer la consulta. L'usuari no la podria fer. A més, el model relacional no considera les **jerarquies d'agregació** (les ciutats s'agreguen en regions, les regions en estats, etc.).

Mecanismes de protecció d'inferència

Entenem per mecanismes de protecció d'inferència tot el que ajudi a evitar que a partir de les dades d'un cert conjunt d'individus sigui possible inferir o deduir les dades corresponents a aquests; per exemple, aconseguir l'edat d'en Jordi a partir de la mitjana d'edat dels habitants de Lleida.

El que necessitem és quelcom més flexible que una base de dades estadística. Després de demanar la suma de vendes per estats, ens hauria de permetre aïllar-ne un (per exemple, el que té la major proporció de vendes en relació amb la seva població) i veure les vendes desglossades per regions. L'usuari final, sense cap coneixement específic d'informàtica, hauria de poder navegar fàcilment per les dades.

Navegar

En aquest context, el terme *navegar* significa fer un conjunt de consultes de manera que quan es vegi el resultat d'una, es decideix quina serà la següent.

Una base de dades relacional (estadística o no) no proporciona la flexibilitat i facilitat d'ús que requereix l'anàlisi en línia.

1.2. Fulls de càlcul

Realment, els analistes no utilitzen les bases de dades estadístiques, principalment pel fet que són difícils d'utilitzar. El problema no és que no es puguin usar per a tasques d'anàlisi, sinó com resulta de difícil i pesat fer-ho. El que sí que utilitzen en el seu dia a dia són els fulls de càlcul, que sense cap mena de dubte són molt més fàcils d'utilitzar. Cada cel·la conté una dada que podem referenciar fàcilment mitjançant les seves coordenades bidimensionals (files × columnes). Podem operar amb les dades simplement referenciant-les amb una lletra i un número. Si una dada canvia, no cal modificar totes les fórmules en les quals apareix, sinó només canviar el valor d'una cel·la concreta. Aquest funcionament facilita el que es denomina anàlisi *what-if* ('què passa si'). És a dir, què passarà si canvio aquest valor (per exemple, un preu de venda, la quantitat d'unitats produïdes, el temps de producció, etc.)? Com variarà el meu negoci?

Cel·les tridimensionals

Si a més de files i columnes, el full de càlcul permet treballar amb pàgines, podem parlar d'un espai de cel·les tridimensional.

Malgrat aquesta facilitat d'ús, els fulls de càlcul encara tenen algunes mancances:

- No són adequats per a grans quantitats de dades.

- No aporten cap significat a les dades (les cel·les s'identifiquen simplement per les seves coordenades).

- La creació d'informes no és prou senzilla.

- De la mateixa manera que les bases de dades estadístiques, no faciliten l'ús de jerarquies d'agregació.

A més d'això, la posició de les dades pot determinar les operacions que es puguin fer. Si no es vol haver d'explicitar una per una tota la llista de cel·les que intervenen en una operació, aquestes han de ser consecutives, de manera que es pugui donar un rang (per exemple, `SUMA(D7:D123)`). Per tant, hem de conèixer *a priori* amb quines dades operarem, per poder col·locar-les en les cel·les adequades. La pregunta que sorgeix en aquest punt és si sempre hi haurà alguna forma de col·locar les dades, de manera que puguem fer fàcilment totes les operacions que vulguem. En els casos en què això no sigui possible, s'hauria de tenir un full de càlcul diferent per a cada operació o conjunt d'operacions que es pugui definir tenint en compte una posició de les dades.

Malgrat que podria solucionar el problema anterior, tenir molts fulls de càlcul amb les mateixes dades generaria molta redundància, amb tots els inconvenients que, com ja sabeu, això suposa. El que ens fa falta és, sense perdre la facilitat d'ús, poder consultar i reestructurar les dades de manera fàcil i flexible.

Necessitem un sistema híbrid que ens proporcioni la flexibilitat i potència d'un full de càlcul, i l'estructura i facilitat de consulta d'una base de dades.

1.3. Eines OLAP i multidimensionalitat

Una de les solucions a les necessitats d'anàlisi de les empreses són les eines OLAP. Aquest terme va ser introduït per E. F. Codd el 1993, tot i que ja hi havia eines informàtiques específiques per a l'anàlisi molta abans. Literalment, fa referència a la possibilitat de processar consultes en línia (en contraposició amb el processament per lots –*batch*–), amb l'objectiu d'analitzar dades. Per entendre millor el que són aquestes eines, podem emprar el denominat test FASMI.[1] Segons aquesta definició, un sistema OLAP ha de fer el següent:

[(1)]De l'anglès *fast analysis of shared multidimensional information*.

Lectura recomanada

Codd, E. F. ; Codd, S. B.; Salley, C. T. (1993). *Providing OLAP (On-line Analytical Processing) to User-Analysts: An IT Mandate*. Arbor Software, Technical Report.

1) FAST: respondre la majoria de les consultes en aproximadament cinc segons (excepcionalment, podria arribar a trigar-ne vint). Això fa que s'hagin d'implementar tècniques específiques d'indexació i cerca, amb mecanismes especials d'emmagatzematge.

El temps de resposta ha de ser petit

Estudis recents demostren que, si un usuari triga més de trenta segons a obtenir el resultat de la seva petició, tendeix a pensar que el procés falla, tret que se l'avisi de la durada. En

qualsevol cas, si la resposta de l'ordinador triga massa a arribar, els usuaris es distreuen i perden el fil dels seus raonaments.

2) ANALYSIS: oferir eines d'anàlisi estadística i generació d'informes sense que calgui programar res (ni tan sols mitjançant un llenguatge de quarta generació –4GL– com l'SQL). Una eina OLAP podria ajudar a l'estudi de sèries temporals, l'adjudicació de costos, el canvi de monedes, la prospecció de dades, la definició de ràtios, etc. Només amb un magatzem de dades ja podríem respondre les preguntes què i qui. Amb una eina OLAP, a més d'això, també hauríem de poder respondre per què i què passa si…? Les eines OLAP són un complement imprescindible dels magatzems de dades.

3) SHARED: implementar els mecanismes de seguretat (control d'accés i confidencialitat de les dades) i concurrència (encara que els analistes no vulguin escriure noves dades, sí que desitjaran escriure i possiblement compartir els resultats de l'anàlisi) necessaris per compartir informació.

4) INFORMATION: ser capaç de guardar tota la informació necessària. Aquest punt té dos vessants. En primer lloc, el volum de dades pot arribar a ser molt gran. D'altra banda, el sistema no s'ha de limitar a contenir dades, sinó que també ha de registrar quin és el seu significat: les metadades (informació = dades + metadades).

Aquestes quatre característiques són molt importants, però encara queda la cinquena, la més important de totes, la que realment distingeix les eines OLAP: la **multidimensionalitat**.

Qualsevol eina OLAP ha de ser multidimensional.

La multidimensionalitat

La multidimensionalitat no té els seus orígens en les bases de dades, sinó que es basa en l'àlgebra de matrius, que ha estat utilitzada per a l'anàlisi manual de dades des del segle XIX.

Com ja hem dit abans, els analistes no són informàtics. Per això, han de ser les eines les que s'adaptin a ells, i no a l'inrevés. Volem que els usuaris no depenguin del departament d'informàtica per fer una simple consulta. Ara no parlem d'un entorn transaccional amb consultes predefinides que gairebé mai no canvien, sinó d'un entorn d'anàlisi amb consultes *ad hoc* que l'usuari ha de formular a mesura que té la necessitat de veure algunes dades. Per facilitar-los la feina, una eina OLAP ha de presentar les dades com els analistes estan acostumats a veure-les, és a dir, en termes de **Fets** i **Dimensions**, en lloc de taules, atributs i claus foranes.

Exemple de Fets i Dimensions d'anàlisi

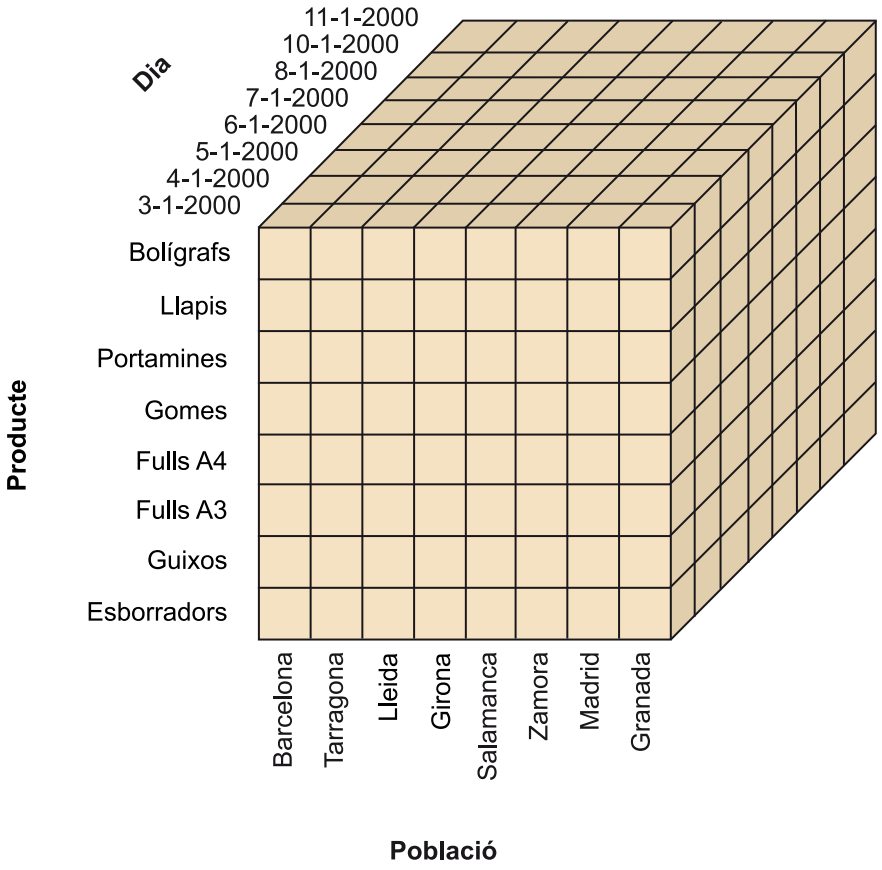
Imaginem-nos que es vol analitzar la distribució i els valors de les vendes d'una cadena de supermercats. Vendes seria el nostre Fet objecte d'anàlisi. Un possible espai tetradimensional per analitzar aquest Fet estaria definit en aquest cas per les Dimensions `Producte`, `Client`, `Temps` i `Poblacio` en la qual s'ha produït la venda.

La multidimensionalitat es basa en la dicotomia entre dades mètriques (què volem analitzar) i dades descriptives (què, a qui, on, quan, com, etc.). Les Dimensions (dades descriptives) defineixen un espai *n*-dimensional, conegut com a **cub**, en el qual col·loquem els Fets (dades mètriques) que volem analitzar (en certa manera, això generalitza els fulls de càlcul, de manera que podem tenir qualsevol nombre de Dimensions). A cada posició en aquest espai la denominarem **cel·la**. Cada cel·la correspon a un Fet concret que queda determinat per les Dimensions d'anàlisi que utilitzem. Observeu la diferència amb els fulls de càlcul: les cel·les no s'identifiquen per simples caràcters muts, sinó pels valors de les Dimensions d'anàlisi, que sí que tenen associat un significat.

Hipercub

Denominar-lo cub és un abús de llenguatge, ja que les Dimensions no necessàriament tindran la mateixa longitud. A més, generalment tindrà més de tres Dimensions. Per tant, hauríem de parlar en tot cas d'hipercub.

Figura 2



Cub tridimensional

Si per simplificar el dibuix oblidem de manera momentània la Dimensió Client, en el cas anterior tindríem un cub com el de la figura 2. Cadascuna de les cel·les d'aquest cub representa vendes, que és el tipus de Fet que volíem analitzar. Concretament, la cel·la que hi ha en la intersecció entre «Lleida», «Gomes» i «7-1-2000» contindrà totes les dades de què disposem sobre les vendes d'aquest article, en aquesta població i en la data proporcionada.

El cub

El cub no és més que una metàfora de com s'han d'entendre les dades. No significa que les eines hagin de dibuixar cubs *n*-dimensionals a la pantalla, ni que forçosament hagin d'emmagatzemar les dades en matrius de *n* Dimensions. Aquest concepte serveix simplement per ajudar als usuaris a entendre el que poden fer amb les dades.

La multidimensionalitat consisteix simplement a concebre les dades que volem analitzar en termes de Fets i Dimensions d'anàlisis, de manera que les podem situar en un espai *n*-dimensional.

Disposar d'una gran base de dades que abasti a tota l'empresa no té gaire valor per als analistes, que molt probablement es veuran sobrepassats pel seu volum i complexitat. És molt més adequat focalitzar en un únic tema (Fet). La primera contribució de la multidimensionalitat és que permet donar a cada analista sol el cub o el conjunt de cubs que corresponguin al Fet o als Fets en els quals estigui interessat. D'aquesta manera, reduïm la complexitat del problema i en simplifiquem la feina.

APL

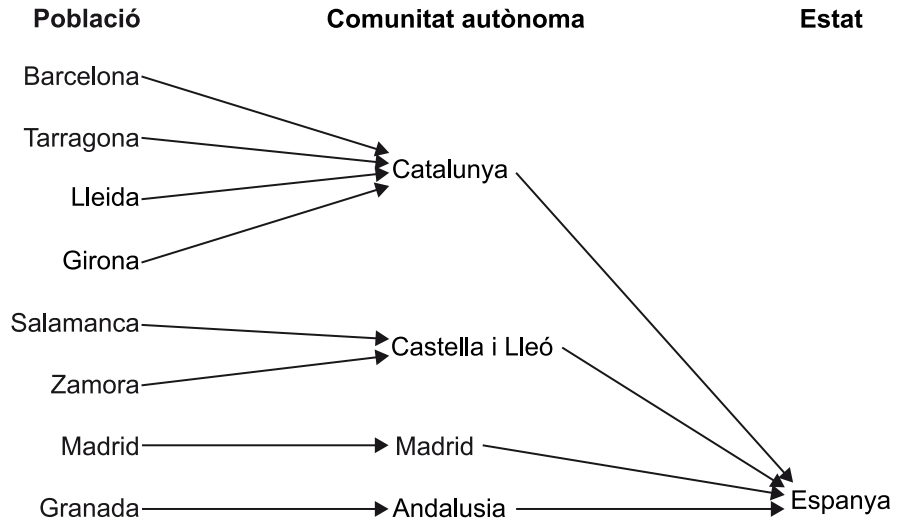
La primera eina multidimensional que corria sobre un ordinador va ser el llenguatge de programació APL, desenvolupat per IBM al final de la dècada dels seixanta. Oferia la possibilitat de definir variables multidimensionals i operar-les mitjançant un conjunt d'operadors específics. Malgrat no ser gaire amigable, es va utilitzar molt durant tota la dècada dels setanta.

Encara que, d'una banda, simplifiquem la visió de les dades per a què els analistes els puguin entendre, per l'altra, hem d'afegir les funcionalitats que demanen. En aquest sentit, hem de parlar de **nivells de detall** i **jerarquies d'agregació**. Els punts que hi ha a cada Dimensió es poden agrupar per nivells segons una certa jerarquia. Un conjunt de punts d'un cert nivell formen un altre punt en el nivell immediatament superior. Això és especialment important perquè, per prendre decisions, el més habitual és mirar les dades resumides o agregades per grups (per exemple, gammes de productes, regions geogràfiques, etc.).

Jerarquia d'agregació de la Dimensió geogràfica

Podem veure un exemple molt clar de jerarquia d'agregació de la Dimensió geogràfica dibuixada a la Figura 3. Un conjunt de ciutats formen part d'una comunitat autònoma i un conjunt de comunitats formen un estat. D'aquesta manera, la nostra jerarquia d'agregació de la Dimensió geogràfica té tres nivells d'agregació diferents: `Poblacio`, `Comunitat i Estat`.

Figura 3

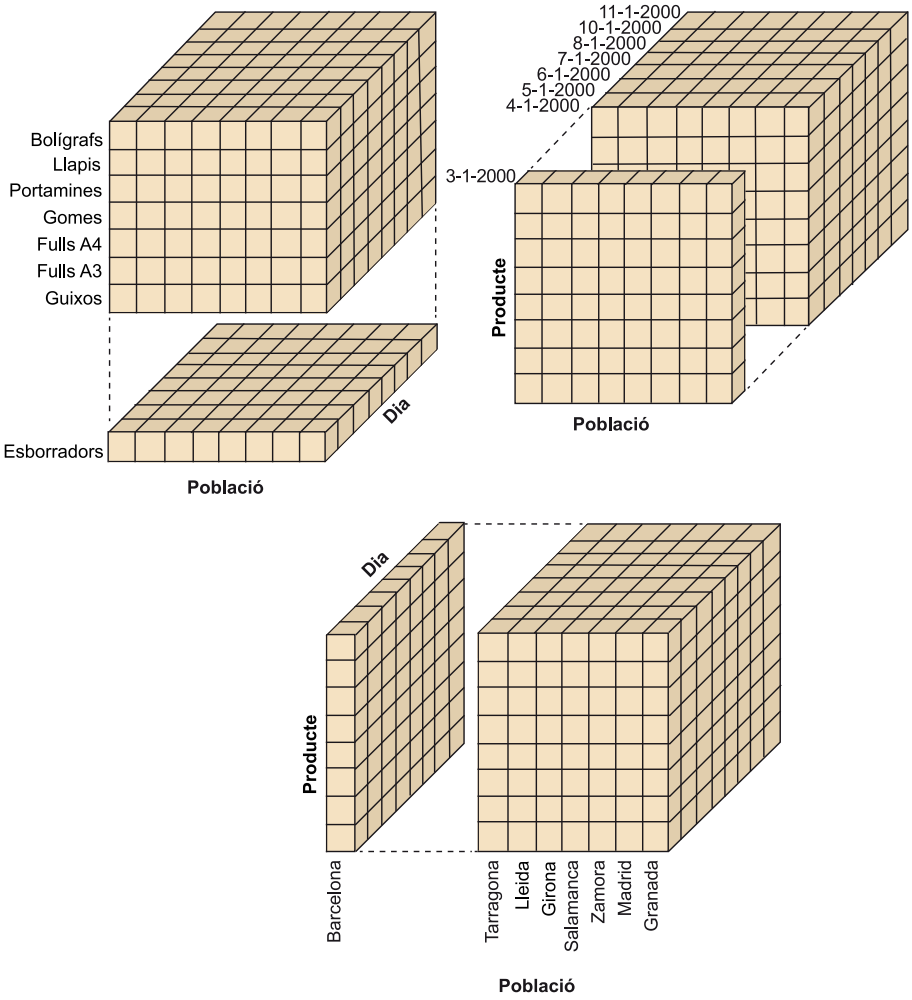


La multidimensionalitat proporciona molt més que la simple possibilitat de visualitzar les dades en forma de cub. També ens proporciona els fonaments per poder manipular-ho de manera fàcil i flexible, sense perdre capacitat de càlcul. El canvi del nivell de detall, amb la possibilitat de seleccionar els elements de les Dimensions i canviar l'objecte d'anàlisi és el que denominarem navegabilitat. Podem continuar veient aquesta navegabilitat en termes de cubs. D'aquesta manera, les eines OLAP ofereixen, amb petites variacions, les operacions que es detallen a continuació.

Slice:[2] fa un tall al cub de manera que es redueix el nombre de Dimensions. Es tracta simplement de fixar un valor a una de les Dimensions del cub, de manera que passem a tenir un cub amb *n*-1 Dimensions en el qual totes les cel·les fan referència al valor que hem triat.

(2)*Slice* significa literalment 'porció'.

Figura 4



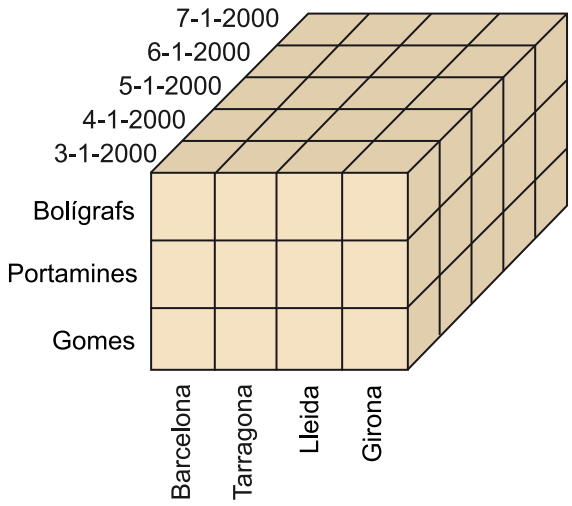
Exemple d'*slice*

La figura 4 exemplifica les tres possibles maneres de fer *slices* en un cub tridimensional. Fixem-nos en la de dalt a l'esquerra. El que fem en aquest cas és, de tot el cub tridimensional que teníem, quedar-nos només amb un cub bidimensional (una *slice*), en el qual totes les cel·les contenen dades referents a «Esborradors». Si a aquest cub bidimensional resultant li tornéssim a fer una *slice*, ara per al valor «Salamanca» en la Dimensió geogràfica, ens quedaria un cub unidimensional (una línia de cel·les), que mostraria les vendes d'esborradors que hi ha hagut a Salamanca cada un dels dies.

Dice: [3] selecciona un subespai del cub original, sense reduir el nombre de Dimensions. Això s'aconsegueix seleccionant un subconjunt de valors en cadascuna de les Dimensions.

(3)Una traducció literal al castellà és 'dau'.

Figura 5



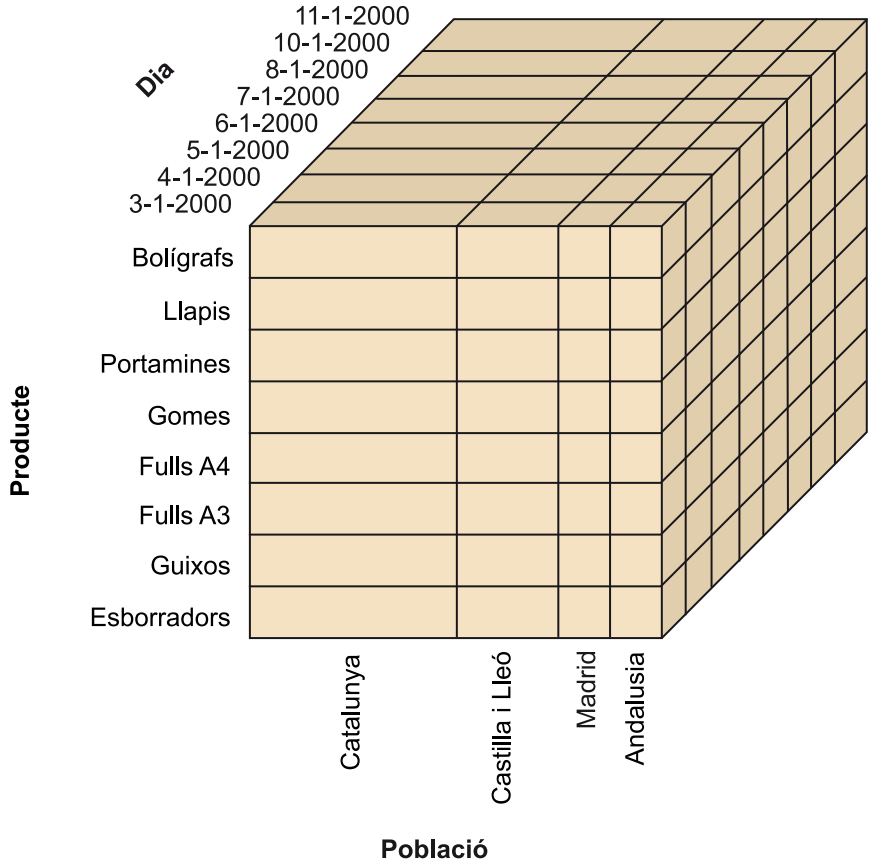
Exemple de *Dice*

A la figura 5, podem veure el resultat d'aplicar una operació de *Dice* al cub tridimensional que podem veure a la Figura 2. De tot l'espai de mida 8 × 8 × 8 que teníem de manera originària, ens quedem només amb un subespai de 4 × 3 × 5, que té moltes menys cel·les, però que continua essent tridimensional.

Roll-up:[4] redueix el detall amb el qual veiem les dades. Agrupa les cel·les del cub seguint una certa jerarquia d'agregació. Per obtenir les dades de cada cel·la, aplica una certa operació matemàtica (com per exemple, la suma o la mitjana) a les dades de les cel·les que formen part de cadascun dels grups.

(4)'Enrotllar', en castellà.

Figura 6



Exemple de *roll-up*

La figura 6 mostra el resultat de fer un *roll-up* fins al nivell `Regio` en el cub que tenim a la figura 2. Ara, en lloc de disposar d'una cel·la per a cadascuna de les ciutats catalanes, tenim una sola cel·la que conté les dades que fan referència a tot el grup de ciutats de Catalunya. El que hem fet és moure'ns des del nivell `Poblacio` fins al nivell `Regio` dins de la jerarquia d'agregació de la Dimensió geogràfica que hem explicat en l'exemple corresponent. Passa el mateix per a les ciutats de Castella i Lleó, Andalusia i Madrid. Aquests dos últims casos són molt senzills, perquè cada regió conté només una ciutat (en el nostre cas) i, per consegüent, les dades de la regió són les mateixes que les de les ciutats corresponents. No obstant això, què passa amb «Catalunya» i «Castella i Lleó»? Com obtenim les seves dades? Hem d'aplicar una certa funció d'agregació. Si parlem de quantitats venudes, la de tota Catalunya és la suma de les quantitats venudes a Barcelona, Tarragona, Lleida i Girona (considerant que només tenim botigues en aquestes ciutats, o que només ens interessen aquestes).

Drill-down:[5] augmenta el detall amb el qual veiem les dades. És l'operació inversa al *roll-up*. En comptes de pujar en una jerarquia d'agregació, baixem i desfem els grups. Observeu que les funcions d'agregació que hem utilitzat en fer *roll-up* no es poden desfer, si no disposem de les dades originals. Heu d'entendre aquesta operació com un desfer (*undo*) de l'operació *roll-up*.

(5)Accés al detall subjacent de les dades.

Exemple de *drill-down*

Aquesta operació correspondria al pas del cub de la figura 6 al de la figura 2. Observeu que per poder fer-ho, hem de disposar de les dades originals. Com aconseguiríem les vendes en cadascuna de les poblacions catalanes si només tinguéssim la suma de quantitats venudes a tota Catalunya?

Drill-across:[6] canvia el tema d'anàlisi. Després d'aplicar aquesta operació, continuem disposant del mateix espai *n*-dimensional que teníem, però ara les cel·les contindran dades que corresponen a un tipus de Fet diferent. En termes d'àlgebra relacional, el *drill-across* s'assembla a una combinació (*join*), en el sentit que associa cada element d'un cub amb un element d'un altre, de la mateixa manera que fa la combinació entre taules.

(6)*Drill-across* literalment significa 'foradar a través'. Realment, s'utilitza per semblança a *drill-down*.

Exemple de *drill-across*

Amb aquesta operació, si partim del cub de la figura 5, obtindrem un cub igual que aquell però que, en lloc de contenir dades de vendes, conté dades de producció, per exemple. La cel·la que hi ha en la intersecció entre «Lleida», «Gomes» i «7-1-2000» contindrà totes les dades que tinguem sobre la producció d'aquest article, en aquesta població i en la data proporcionada.

Les Dimensions s'utilitzen per seleccionar i afegir les dades al nivell de detall desitjat.

Aquestes operacions, juntament amb el canvi d'ordre dels elements que formen les Dimensions i el canvi de posicions d'aquestes (dit d'una altra manera, fer una rotació o pivotar), són millores importants respecte a la rigidesa en les files i columnes d'un full de càlcul. No obstant això, deixant de banda la navegabilitat i aquesta flexibilitat de presentació, les eines OLAP també han d'oferir facilitats per fer informes. Encara que en pantalla fossin capaços de representar tres o més Dimensions, en paper això resulta clarament complicat. Podeu trobar representacions gràfiques més o menys sofisticades, però podríem dir que les dues possibilitats de representació més habituals són les taules estadístiques (com, per exemple, les de la taula 1) i les taules relacionals (per exemple, les de les taules 2 i 3).

Taula 2

Dia	Producte	Població	Nombre d'articles	Ingressos
5-1-2000	Bolígrafs	Barcelona	15	39,5
5-1-2000	Gomes	Barcelona	3	1,4
5-1-2000	Portamines	Barcelona	4	8,7
5-1-2000	Bolígrafs	Tarragona	3	5,1
5-1-2000	Gomes	Tarragona	1	0,3
5-1-2000	Portamines	Tarragona	0	0
5-1-2000	Bolígrafs	Lleida	7	15,9
5-1-2000	Gomes	Lleida	0	0
5-1-2000	Portamines	Lleida	6	11,0
5-1-2000	Bolígrafs	Girona	1	1,2
5-1-2000	Gomes	Girona	5	1,4

Dia	Producte	Població	Nombre d'articles	Ingressos
5-1-2000	Portamines	Girona	2	5,1

Taula 3

Dia	Producte	Població	Nombre d'articles	Ingressos
5-1-2000	Bolígrafs	Catalunya	26	61,7
5-1-2000	Gomes	Catalunya	9	3,1
5-1-2000	Portamines	Catalunya	12	24,8

Exemple d'informe

Pensem que, del cub de la figura 5, només ens quedem amb la part que conté les dades del dia «5-1-2000». A més, imaginem-nos que cada cel·la conté tant el nombre d'unitats venudes com la quantitat d'euros ingressada. A la taula relacional de la taula 2, podeu veure l'informe que resultaria d'aquesta consulta. Si ara féssim un *roll-up* fins al nivell `Regio`, obtindríem l'informe de la taula relacional de la taula 3, en la qual tenim resumides les dades per a tot Catalunya (hem sumat les dades de les ciutats).

La simplicitat de la concepció multidimensional de les dades suposa dos beneficis. D'una banda, ajuda als analistes a entendre les dades. De l'altra, ajuda als informàtics a preveure les consultes que faran els analistes, de manera que pot optimitzar el temps de resposta amb molta més facilitat.

2. Components del model multidimensional

Ara que ja heu vist què és una eina OLAP i els conceptes bàsics de la multidimensionalitat, ho formalitzarem estudiant per separat les estructures, operacions i restriccions d'integritat pròpies d'un model multidimensional. Aquest model és independent de qualsevol eina i us servirà per aclarir els conceptes generals. Podeu considerar que el que veureu en aquest apartat suposa per a les eines OLAP el mateix que el model relacional suposa per a les bases de dades relacionals.

Els tres components d'un model de dades

Qualsevol model de dades està format per tres components: estructures de dades, operacions sobre les dades i restriccions d'integritat inherents al mateix model. En el cas del model relacional, les estructures són les relacions; els conjunts d'operacions, per exemple, l'àlgebra relacional o el llenguatge SQL; i les restriccions d'integritat, la unicitat i entitat de la clau primària, la integritat referencial i la integritat de dominis.

2.1. Estructures de dades

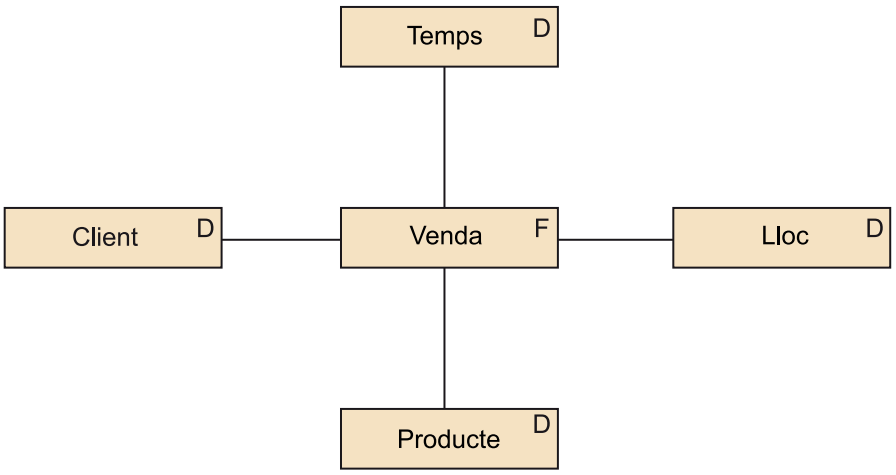
El model multidimensional està marcat per la dicotomia Fet-Dimensió. Tots els elements han d'estar a un costat o a l'altre d'aquesta línia divisòria. Per consegüent, el primer que hi ha són Fets i Dimensions.

Un Fet representa un tema objecte d'anàlisi.

Una Dimensió representa un punt de vista que utilitzarem en l'anàlisi de les dades.

Per dibuixar els elements i les relacions entre ells, utilitzarem la notació d'UML. Dimensions i Fets són classificadors. Per tant, els dibuixarem amb rectangles amb el nom en el seu interior, però posarem una *D* o una *H* a la cantonada superior dreta per distingir els uns dels altres. Relacionarem cada Fet amb les seves Dimensions mitjançant associacions.

Figura 7



Exemple de Fets i Dimensions

A la figura 7, podeu veure un esquema amb un Fet i quatre Dimensions d'anàlisi. Volem analitzar les vendes segons quan i on es van produir, i què i a qui es va vendre.

Els classificadors (Fets i Dimensions) contenen altres elements que proporcionen més detalls sobre les dades. Comencem primer per analitzar el contingut de les Dimensions.

2.1.1. Dimensions

Sabem que una Dimensió representa un punt de vista des del qual es poden analitzar les dades. Considerarem una Dimensió específica i a partir d'aquí definirem els conceptes.

Dimensió

Dimensió ve del llatí *dimetiri*, que es tradueix per 'mesurar' (per exemple, diem que les dimensions d'una nevera són 60 × 60 × 180). Realment, els romans van agafar prestada la paraula dels grecs, que la utilitzaven en un sentit molt més filosòfic, equivalent a l'accepció que trobem actualment en la geometria (per exemple, parlem d'espai tridimensional).

Dins d'una Dimensió, podem distingir grups d'instàncies segons la seva mida (granularitat).

Un Nivell representa un conjunt d'instàncies d'una Dimensió que tenen la mateixa granularitat, i el dibuixarem com una classe (un rectangle amb els noms a dins) amb una *N* a la cantonada superior dreta.

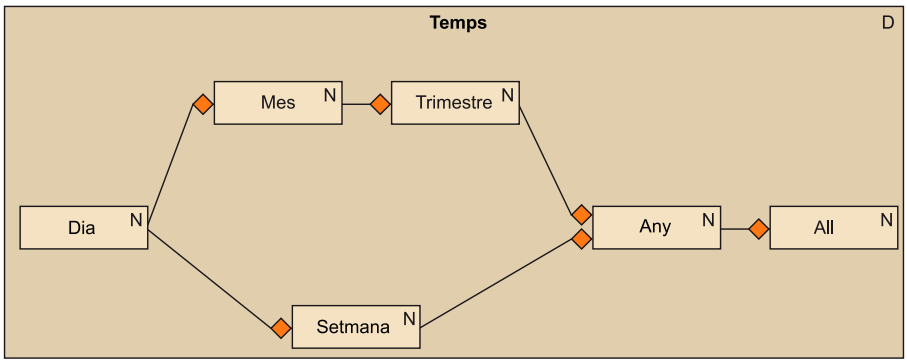
Exemple de granularitats

Dins de la Dimensió `Lloc`, tindríem que les poblacions posseeixen una granularitat més petita que les regions i aquestes tenen una granularitat més petita que els estats. Per tant, representarem amb una mateixa classe (Nivell) totes les poblacions, però tindrem una classe diferent per a les regions i una altra més per als estats. Les tres classes representen llocs i, per tant, estaran dins de la mateixa Dimensió.

Les instàncies d'un cert Nivell s'agrupen per donar lloc a instàncies d'un altre Nivell de granularitat més gran. Podem dir que les instàncies d'un Nivell formen instàncies d'un altre Nivell, o que hi ha relacions part-tot entre Nivells.

Representarem aquestes relacions amb agregacions entre els Nivells i distingirem un Nivell especial, que denominarem `All`, i que representa l'agrupació de totes les instàncies de la Dimensió al mateix temps.

Figura 8



Exemple de jerarquia d'agregació no lineal

En la Dimensió temporal dibuixada a la figura 8, podem veure que un conjunt de dies dona lloc a un mes, però un conjunt de dies també pot donar lloc a una setmana. També podem agrupar mesos per obtenir trimestres i podem obtenir anys per agrupació de setmanes o trimestres. Finalment, veiem un Nivell especial `All` que representa el grup format per tots els anys que estem interessats a analitzar.

Generalment, els Nivells dins d'una Dimensió i les agregacions que els uneixen formen un graf dirigit, conegut com a jerarquia d'agregació.

Dels axiomes de la mereologia es poden deduir les següents propietats d'aquest graf.

Mereologia

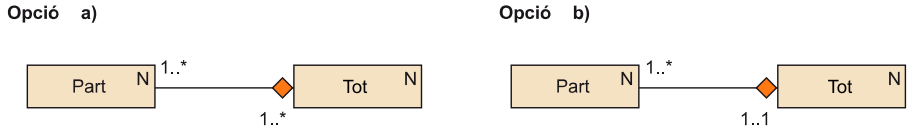
La mereologia és la ciència que estudia les relacions part-tot.

- No pot contenir cicles.
- Conté un únic Nivell que no té parts (el denominen atòmic).
- Pot haver-hi o no un Nivell `All`, però si hi és:
  - Només n'hi ha un.
  - Conté exactament una instància.
  - No és part de cap altre Nivell.
- Tots els Nivells que no són part de cap altre Nivell es poden connectar directament al Nivell `All`.
- Totes les instàncies (menys les del Nivell atòmic) han de tenir almenys una part.
- Totes les instàncies (menys les del Nivell atòmic) poden tenir més d'una part.

- Els conjunts de parts de dues instàncies d'un mateix nivell no han de ser necessàriament disjunts.

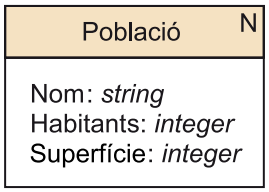
- Sempre podem construir el graf de manera que totes les instàncies participin en un tot, menys la del Nivell `All`.

Figura 9



La única elecció que hi ha realment és si una part pot participar en un únic tot (opció b) o en més d'un (opció a). El més habitual és l'opció a; per tant, si no s'indica res, pensarem que la multiplicitat que tenim és aquesta.

Figura 10



Com podeu veure a la figura, un Nivell té associats un conjunt d'atributs, els denominats Descriptors.

Els atributs que podem trobar en un Nivell contenen la informació no jeràrquica i estan definits sobre un domini discret. Es denominen Descriptors.

Els Descriptors no s'utilitzen per formar grups, sinó simplement per seleccionar instàncies o mostrar-los en els informes.

Relacionant els nous conceptes, veiem el següent:

Una Dimensió conté un conjunt de Nivells relacionats per agregacions. Cada un dels Nivells té atributs que denominem Descriptors.

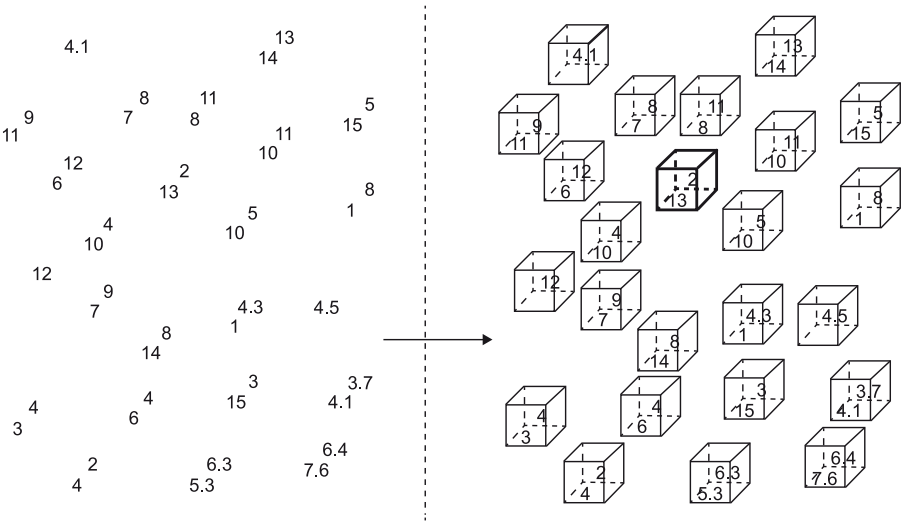
Finalment, malgrat que l'anàlisi multidimensional es basa en l'àlgebra de matrius, cal dir que, al contrari que en el cas dels espais lineals, el model multidimensional, per si mateix, no inclou cap mena d'ordre o distància entre les instàncies d'una Dimensió. Si parléssim de la Dimensió temporal, podríem definir fàcilment aquest ordre (les deu del matí van abans que les tres de la tarda i estan separades per cinc hores). No obstant això, com ho faríem amb la Dimensió de productes o clients?

2.1.2. Fets

Oblidem-nos per un moment de les Dimensions i pensem que volem analitzar els Fets. Tenim un conjunt immens de dades (que denominarem **mesuraments**, en un sentit ampli de la paraula) dins del nostre magatzem de dades. El que volem és posar una mica d'ordre en aquest conjunt inabastable i acostar les dades als analistes. El primer que cal fer és representar els mesuraments que fan referència al mateix esdeveniment[7] dins d'una mateixa estructura. Denominarem a això cel·la (escrit amb minúscules).

[7] Qualsevol succés o concepte susceptible de ser analitzat.

Figura 11



Exemple de cel·les

A la part esquerra de la figura 11 tenim un conjunt de mesuraments: quantitats produïdes, quantitats venudes, costos, ingressos, etc. A la part dreta de la mateixa figura, hem agrupat els mesuraments que fan referència al mateix esdeveniment. Per exemple, la cel·la en negreta podria correspondre a la venda que es va fer a en Jordi fa un parell de dies. Li vam vendre dos objectes i li'n vam cobrar tretze euros.

Malgrat aquesta primera agrupació de mesuraments en cel·les, encara no complim els requeriments dels usuaris. El pas següent és poder unir cel·les per obtenir-ne d'altres de més grans, que no només representin un esdeveniment, sinó molts alhora (per exemple, totes les vendes que es van fer abans-d'ahir: la d'en Jordi, la d'en Joan, la d'en Pere, etc.). El conjunt de totes les cel·les C amb la unió $\cup$ forma un semigrup commutatiu. És a dir, compleix les propietats següents:

- Tancat (la unió de dues cel·les sempre és una altra cel·la).

$$\forall x,y \in \mathrm{C}\ x \cup y \in \mathrm{C}$$

- Commutatiu (no importa l'ordre en què fem una unió: obtindrem la mateixa cel·la).

$$\forall x,y \in C\ x \cup y = y \cup x$$

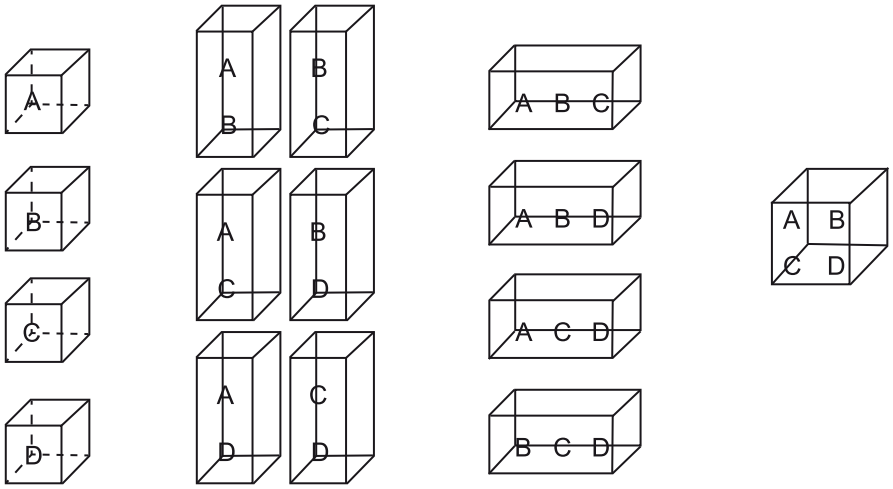
- Associatiu (podem prioritzar una seqüència d'unions com vulguem, sense alterar la cel·la resultat).

$$\forall x,y,z \in C\ x \cup (y \cup z) = (x \cup y) \cup z$$

- Element neutre (hi ha un element que, operat amb qualsevol altre, no el modifica).

$$\forall x \in C\ x \cup \varnothing = x$$

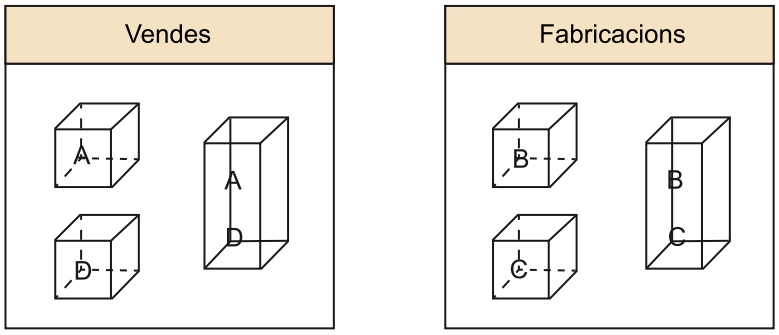
Figura 12



Si denominem CA al conjunt de totes les cel·les atòmiques[8] i permetem totes les unions possibles, obtenim que C conté $2^{Card(CA)}$ o $2^{Card(CA)-1}$ cel·les (que és la cardinalitat del conjunt de parts de CA, Card(P(CA)). A la figura 12, podeu veure totes les cel·les que es poden obtenir a partir de quatre cel·les atòmiques. La cardinalitat de C creix de manera exponencial respecte al nombre de cel·les atòmiques. Per consegüent, no cal tenir gaires cel·les atòmiques perquè el problema de guardar o consultar C sigui intractable.

[8] Les cel·les atòmiques són aquelles que no podem obtenir com a resultat de la unió amb altres cel·les.

Figura 13



Pensem ara en el significat d'aquesta unió de cel·les. Quin sentit té fer la unió d'una cel·la que representa vendes amb una cel·la que representa la fabricació d'un cert producte? Quin tipus de cel·la obtindrem com a resultat? Una venda? Una fabricació? No té gaire sentit unir cel·les de tipus diferents. Si restringim la unió a cel·les del mateix tipus (el mateix Fet), ja només tenim $\sum i$ Card(P(CAi)) cel·les, essent CAi el conjunt de cel·les atòmiques instància del Fet i. A la figura 13, podeu veure quines cel·les podem obtenir si considerem que A i D són d'un tipus de Fet, i B i C d'un altre tipus.

El conjunt de cel·les s'ha reduït de manera dràstica. No obstant això, encara n'hi ha moltes que no interessen als analistes. Les cel·les guanyen significat sol quan estan associades a instàncies de les Dimensions. Únicament quan sabem quin producte es va vendre, a qui el vam vendre, on el vam vendre, etc. la cel·la aconsegueix tot el seu significat. Per tant, només interessen les cel·les que estan vinculades a una instància de cadascuna de les Dimensions. No agrupem qualsevol conjunt de cel·les, sinó que utilitzem com a criteri d'agrupació les Dimensions. No agruparem una cel·la que contingui dades mensuals amb una altra que contingui dades trimestrals. A quin nivell de la Dimensió temporal hi hauria el resultat d'aquesta unió? El que sí que farem és agrupar tres cel·les mensuals per obtenir una cel·la trimestral.

Les instàncies de les Dimensions

Recordeu que les instàncies de les Dimensions estan classificades per Nivells. Per tant, cada cel·la està associada a un Nivell en cadascuna de les Dimensions.

Al conjunt de cel·les del mateix Fet que estan associades a instàncies del mateix Nivell per a cadascuna de les Dimensions, el denominarem Cel·la.[9]

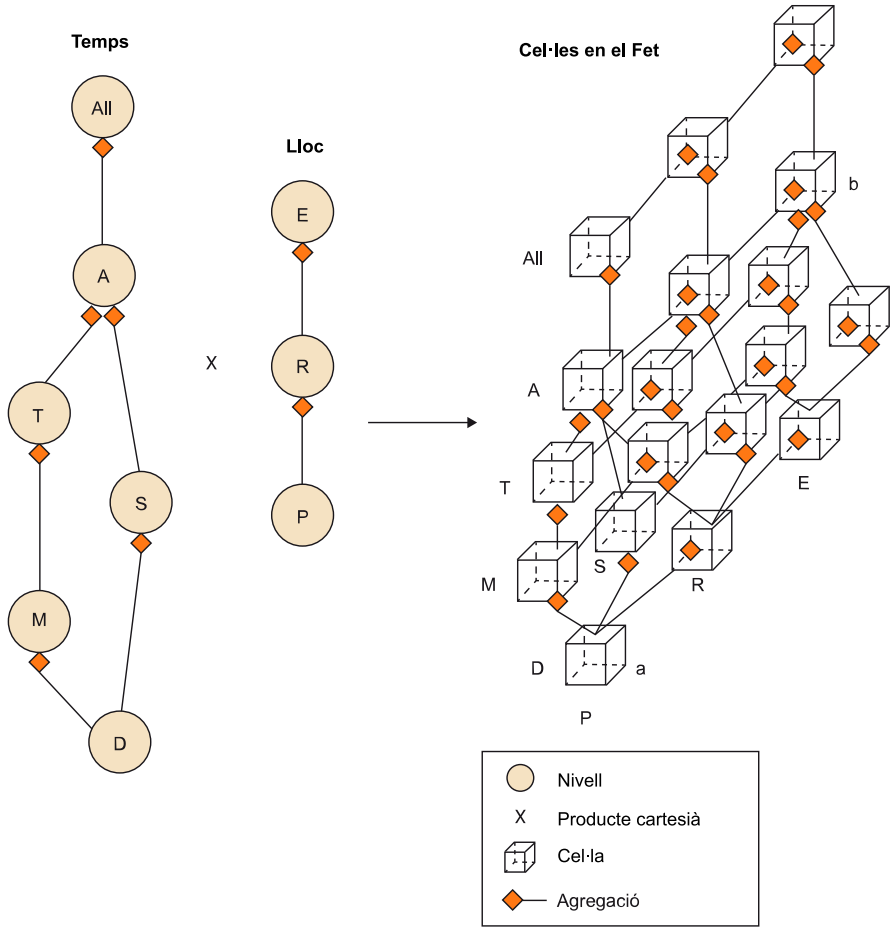
[9]Utilitzarem la majúscula per distingir els conjunts de cel·les de les cel·les individuals.

Una Cel·la (que dibuixarem com una classe amb una *C* a la cantonada superior dreta) representa un conjunt d'instàncies d'un Fet que tenen la mateixa granularitat.

Dins d'un Fet, tindrem tantes Cel·les com elements hi ha en el producte cartesià dels Nivells de les Dimensions. Igual que els Nivells, aquestes Cel·les estaran relacionades per agregacions. Això indica que les instàncies d'una Cel·la componen les instàncies de la Cel·la immediatament superior.

Les instàncies de les cel·les s'agrupen en **jerarquies d'agregació**.

Figura 14

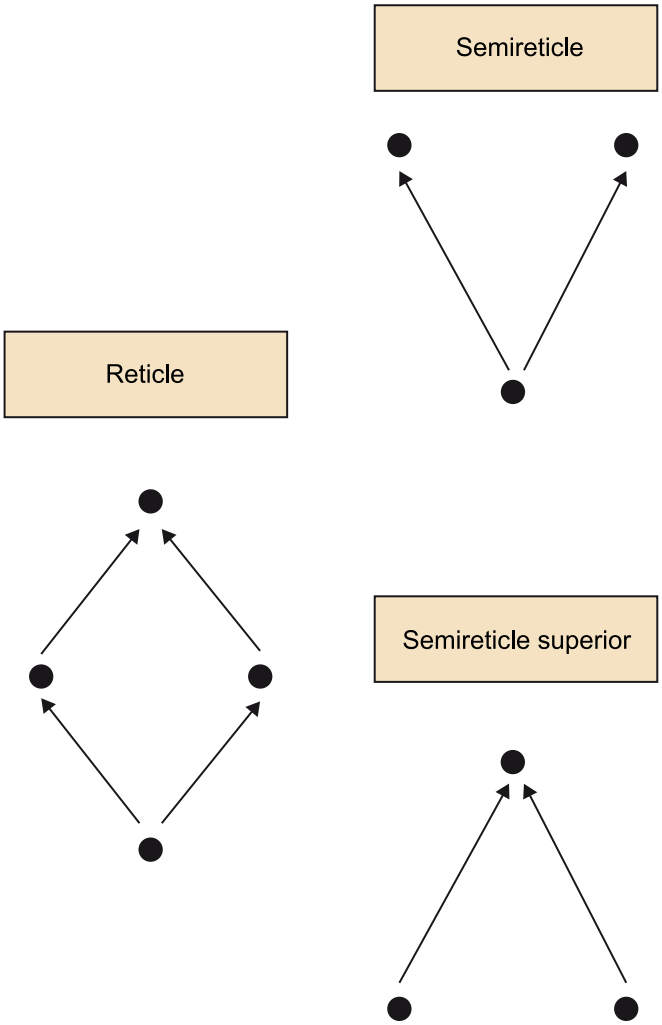


Exemple de Cel·les en un Fet

A la figura 14, teniu dibuixat un esquema amb les dues Dimensions que hem vist fins ara: `Temps` i `Lloc`. La primera té sis Nivells i la segona només en té tres. Per a cada combinació possible de Nivells d'aquestes Dimensions, tenim una Cel·la diferent en el nostre Fet. Per exemple, la Cel·la «a» correspon a les granularitats `Dia-Poblacio`, la Cel·la «b» correspon al Nivell *Any-Estat*, i així fins arribar a les divuit (sis per tres) combinacions que hi ha. Totes les instàncies de la Cel·la «b» estan associades amb una instància d'Any i amb una altra d'`Estat`.

Amb això, obtenim que cada Fet conté un graf amb estructura de semireticle inferior, que serà un reticle sol si totes les Dimensions també ho són. Aquest graf ens mostra com podem agrupar les cel·les per obtenir-ne d'altres de més complexes.

Figura 15



Reticle

És un conjunt ordenat en el qual dos elements qualssevol en tenen un altre de suprem (la més petita de les fites superiors o elements majorants) i un altre d'ínfim (la més gran de les fites inferiors o elements minorants). Si només hi ha una de les dues fites, es denomina semireticle (superior o inferior, depenent del que tinguem).

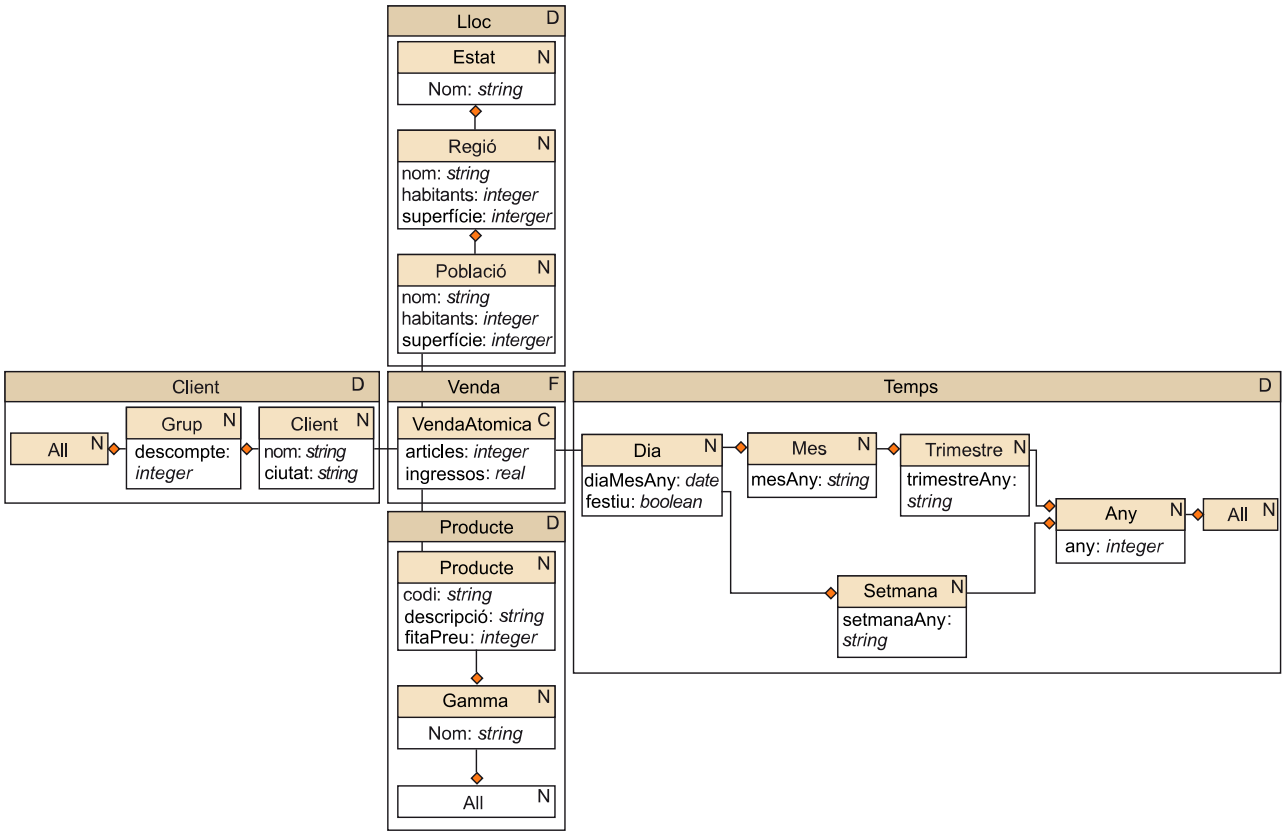
Una Mesura és un atribut d'una Cel·la.

Els valors de les Mesures (que abans hem denominat mesuraments) d'una cel·la s'obtenen com a funció dels mesuraments de les cel·les que la componen. Al contrari del que passa amb els Descriptors, la major part de les Mesures són de tipus numèric, malgrat que algunes són booleanes o, fins i tot, textuals.

Un cop arribats en aquest punt, podem relacionar els conceptes vistos en aquest apartat:

Un Fet conté un conjunt de Cel·les (amb *C* majúscula) relacionades per agregacions. Cadascuna de les Cel·les té atributs que denominem Mesures.

Figura 16



Exemple d'esquema multidimensional

La figura 16 mostra un esquema multidimensional sencer. Per la forma que tenen, se solen denominar esquemes en estrella. Per simplificar-ho, només es dibuixen les Cel·les que tenen especial interès (en aquest cas, només la Cel·la associada al Nivell atòmic de cada Dimensió).

Com podeu veure a la figura 16, associem les Cel·les amb els Nivells corresponents. La multiplicitat d'aquestes associacions sempre és *–1. Cada cel·la s'associa amb una única instància d'un Nivell. Per consegüent, no cal explicitar aquesta multiplicitat.

Els principals elements d'un esquema multidimensional són, d'una banda, les Dimensions, els Nivells i els Descriptors; i de l'altra, de manera simètrica, els Fets, les Cel·les i les Mesures.

2.2. Operacions sobre les dades

En aquest apartat, veurem un conjunt d'operacions algebraiques sobre cubs. Tot i que hem dit que la multidimensionalitat es fonamenta en la representació de les dades en forma de cubs, encara no hem vist què és un cub. Un cub és una funció injectiva que va d'un espai *n*-dimensional finit (definit pel producte cartesià de *n* Nivells $\{N_1, \ldots, N_n\}$) al conjunt d'instàncies d'una Cel·la (C_c).

$$c(x)\colon N_1 \times .. \times N_n \rightarrow Cc$$

Un cub simplement és una funció que diu quina cel·la va a cada punt de l'espai. Per tant, amb ell podem fer qualsevol cosa que faríem amb una funció (per exemple, compondre o unir-ne dos dels mateixos).

Taula 4

Fet ⇒ *Drill-across*	Dimensió ⇒ *CanviBase*
Cel·la ⇒ –	Nivell ⇒ *Roll-up*
Mesura ⇒ *Projecció*	Descriptor ⇒ *Selecció*

Cada operació afecta directament a un sol tipus d'element del model (podeu veure les correspondències a la taula de la taula 4): *drill-across* permet seleccionar un Fet; *CanviBase* permet seleccionar el conjunt de Dimensions que utilitzarem; *roll-up* permet seleccionar el Nivell de detall a cada Dimensió; *Projecció* permet triar les Mesures que volem veure, i *Selecció* fa que puguem utilitzar els Descriptors per triar les instàncies concretes

de cada Nivell que volem veure. Per què no hi ha cap operació associada a Cel·la? Com podem triar la Cel·la que volem veure? No pot haver-hi una operació associada a Cel·la perquè entraria en conflicte amb el *roll-up*. Com que la Cel·la queda determinada pels Nivells que triem, no podem triar Nivells i Cel·les alhora.

Vegem les definicions d'aquestes cinc operacions.

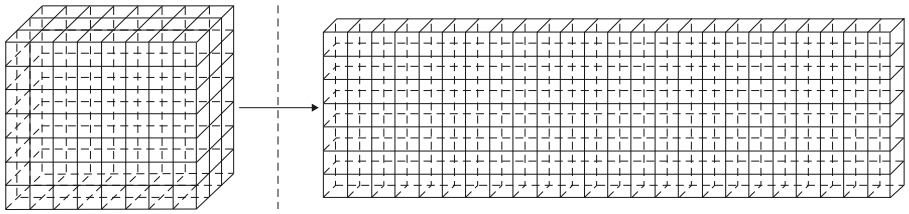
Drill-across: canvia l'objecte d'anàlisi, és a dir, el Fet. Per poder-lo fer, hi ha d'haver alguna relació entre el Fet que tenim i el Fet que volem tenir. Necessitem una funció injectiva entre tots dos, que determini per quina cel·la del cub destí hem de substituir cada cel·la del cub origen. El cas més comú és que els dos Fets comparteixin Dimensions de manera que els mateixos Nivells identifiquin les cel·les dels dos cubs.

$$c_{\text{destí}}(x) = \textit{Drill-across} f(c_{\text{origen}}) = f(c_{\text{origen}}(x))$$

Projecció: simplement selecciona el conjunt de Mesures que volem veure d'entre les que hi ha disponibles a la Cel·la del cub origen. Equival a l'operació homònima de l'àlgebra relacional.

$$c_{\text{destí}}(x) = \textit{Projecció}_{\text{m1,...,mk}}(c_{\text{origen}}) = c_{\text{origen}}(x)[m_1,\ldots,m_k]$$

Figura 17



CanviBase: redistribueix el mateix conjunt de cel·les dins d'un altre espai. Per aplicar-lo, necessitem disposar d'una funció injectiva entre els dos espais (és a dir, entre Dimensions), de manera que cada punt de l'espai del cub destí determini un punt del cub origen. Això pot servir simplement per reordenar els punts de l'espai, o també pot canviar el nombre de Dimensions (com està esquematitzat a la figura 17), però no canvia el nombre de cel·les, ni el seu contingut.

$$c_{\text{destí}}(x) = CanviBase_{\text{f}}(c_{\text{origen}}) = c_{\text{origen}}(f(x))$$

Roll-up: agrupa les cel·les d'un cub en funció d'una jerarquia d'agregació. Augmenta la granularitat fins a un cert Nivell. Redueix el nombre de cel·les, però no el de Dimensions.

$$c_{destí}(x) = Roll-up_{Nivell}(c_{origen}) = \bigcup_{r(y)=x} c_{origen}(y)$$

Selecció: mitjançant un predicat lògic sobre els Descriptors de Nivells, selecciona un conjunt de punts dins de l'espai *n*-dimensional. És absolutament equivalent a l'operació homònima de l'àlgebra relacional.

$$c_{destí}(x) = Selecció_{predicat}(c_{origen}) = \begin{cases} c_{origen}(x), \text{ si } predicat(x) = cert \\ indefinit, \text{ si } predicat(x) = fals \end{cases}$$

Aquest conjunt d'operacions és tancat, és a dir, els operands són cubs i el resultat també. Això implica que podem concatenar les operacions. Per exemple, l'operació *Slice* que hem vist en l'apartat «Eines OLAP i multidimensionalitat» seria equivalent a triar un punt mitjançant l'operació *Selecció* i fer un canvi de base per reduir a un el nombre de Dimensions. Observeu que la Dimensió que eliminem amb el canvi de base només conté la instància «k» i, per tant, tot i que perdem una Dimensió, no perdem cap cel·la (l'espai a × b × 1 té el mateix nombre de punts que l'a × b).

$SliceNi{=}k(corigen) = CanviBasef{:}N1\times..\times Nn \rightarrow N1\times..\times Ni\text{-}1\times Ni{+}1\times..\times Nn(SelecciónNi{=}k(corigen))$

Una altra operació que podeu trobar a faltar és *drill-down*. Com ja hem dit, es tracta de la inversa de *roll-up* i no es pot fer si no disposeu de les dades detallades.

Exemple de seqüència d'operacions

Tenim el cub 2 × 2 × 2 següent:

Unitats produïdes per producte, fàbrica i mes	Fàbrica del Vallès		Fàbrica del Prat	
	Gener 2002	Febrer 2002	Gener 2002	Febrer 2002
Bolígrafs	100.000	110.000	450.000	420.000
Gomes	337.000	473.000	904.000	995.000

Volem veure, per als mateixos mesos i productes, els articles venuts a Catalunya. Hauríem de fer un *drill-across* cap a `Vendes`, però no és possible fer-lo directament perquè les Dimensions no coincideixen. En primer lloc, ens hem de desfer de la Dimensió `Fàbriques`.

A := $Roll\text{-}up_{Fàbrica::All}$("Unitats produïdes per Producte, Fàbrica i Mes")

A	All	
	Gener 2002	Febrer 2002
Bolígrafs	550.000	530.000
Gomes	1.241.000	1.468.000

Ara només tenim un valor en la Dimensió de fàbriques. Per tant, podem fer un canvi de base per quedar-nos amb les mateixes quatre cel·les, però una Dimensió menys.

B := $CanviBase_{Producte\times Temps}(A)$

B	Gener 2002	Febrer 2002
Bolígrafs	550.000	530.000
Gomes	1.241.000	1.468.000

Com que B només té dues Dimensions i les dues les tenim a `Vendes`, ja podem fer el *drill-across*. Com que la funció que utilitzem és la identitat entre Dimensions, no cal explicitar-la, sinó simplement dir quin és el Fet destí (`Vendes` en aquest cas).

C := $Drill\text{-}across_{Vendes}(B)$

C	Gener 2002	Febrer 2002
Bolígrafs	articles: 465.837	articles: 513.284
	ingressos: 973.427'30	ingressos: 1.075.143'80
Gomes	articles: 1.348.378	articles: 1.490.281
	ingressos: 498.462'20	ingressos: 523.093'90

Ara hem obtingut cel·les de `Vendes` amb totes les seves Mesures. Només estàvem interessats en veure articles, així que podem projectar únicament aquesta Mesura.

$$D := Projecció_{articles}(C)$$

D	Gener 2002	Febrer 2002
Bolígrafs	465.837	513.284
Gomes	1.348.378	1.490.281

Ara ja tenim les dades que desitjàvem, però no volem que facin referència a tots els supermercats, sinó només als que hi ha situats a Catalunya. Per tant, cal introduir la Dimensió geogràfica.

$$E := CanviBase_{Articles \times Lloc \times Temps}(D)$$

E	Estat espanyol	
	Gener 2002	Febrer 2002
Bolígrafs	465.837	513.284
Gomes	1.348.378	1.490.281

Si ara volem seleccionar les dades de Catalunya, abans hem d'aconseguir les dades de l'Estat espanyol detallades per regió. En aquest cas ho podem fer, perquè tenim disponibles les dades per població (tal com indica l'esquema de la figura 16). Observeu que si els tinguéssim només emmagatzemats per Estat, no ho podríem fer.

$$F := Drill\text{-}down_{Lloc::Regió}(E)$$

F	Catalunya		Castella i Lleó		Madrid		Andalusia	
	Gener 2002	Febrer 2002	Gener 2002	Febrer 2002	Gener 2002	Febrer 2002	Gener 2002	Febrer 2002
Bolígrafs	275.827	290.918	85.472	111.291	58.172	59.723	46.366	51.352
Gomes	784.172	918.012	293.829	288.409	141.003	140.298	129.374	143.562

Finalment, només hem de seleccionar les dades de Catalunya per tenir el que ens interessa.

$$R := Selecció_{Regió.nom="Catalunya"}(F)$$

R	Catalunya	
	Gener 2002	Febrer 2002
Bolígrafs	275.827	290.918
Gomes	784.172	918.012

2.3. Restriccions d'integritat inherents al model

Ja hem vist quins són els elements del model i les operacions que es poden fer amb les dades. Ara hem de veure quines dades no es poden inserir i quines operacions no estan permeses.

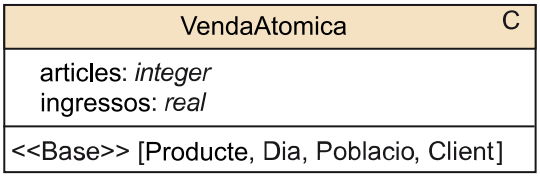
2.3.1. Unicitat i entitat de la Base

Recordeu que un cub és una funció que va d'un espai *n*-dimensional a una Cel·la.

Als diferents conjunts de Nivells que defineixin espais en els quals puguem col·locar les instàncies d'una Cel·la els denominarem Bases.

No cal que totes les Dimensions d'una Cel·la participin en una Base. La Base simplement indica quines Dimensions identifiquen les cel·les (és el mateix concepte que la clau candidata del model relacional).

Figura 18



Exemple de Base

En el nostre exemple de la cadena de supermercats, una certa venda estaria identificada per un dia, un producte, una població i un client. Per tant, `Producte`, `Dia`, `Poblacio` i `Client` defineixen un espai en el qual podem col·locar les instàncies de VendesAtomiques. A la figura 18, podeu veure com ho representaríem.

Les Bases han de complir les restriccions següents:

- No podem col·locar dues cel·les en el mateix punt de l'espai. Pel tant, els valors que tinguin les cel·les per a les associacions amb els Nivells d'una Base han de ser diferents per a cada cel·la. Per exemple, no hi poden haver dues instàncies de `VendaAtomica` associades al mateix `Producte`, `Dia`, `Poblacio` i `Client`.

- Hem de saber en quin punt de l'espai col·loquem cada cel·la. D'aquesta manera, les associacions amb els Nivells d'una Base no admetran el valor nul. Cada cel·la ha d'estar relacionada amb una instància de cada Nivell que forma una Base. Les associacions amb aquests Nivells tenen com a multiplicitat mínima un 1 del costat del Nivell.

- Els Nivells que formen una Base han de ser funcionalment independents. Si hi ha un Nivell que depèn dels altres, el traurem del conjunt que forma la Base. En el nostre exemple, no podria ser que un client sol pogués comprar a una certa població. Si fos d'aquesta manera, la base estaria formada només pels Nivells `Client`, `Dia` i `Producte`. La població ja quedaria determinada pel client.

El conjunt de Nivells que formen una Base han de ser funcionalment independents. A més, les associacions d'aquests Nivells amb la Cel·la han de tenir multiplicitat mínima 1 del costat del Nivell i no és possible que dues cel·les estiguin associades amb les mateixes instàncies per a tots els Nivells de la Base.

2.3.2. Acumulació o agregació

Sens dubte, l'operació més característica de l'anàlisi multidimensional és el *roll-up*. Per desgràcia, també és la més problemàtica. Vegem quines són les tres condicions per a què el resultat d'una agregació sigui correcte.

Compatibilitat

Habitualment, quan s'acumulen un conjunt de dades, el que es fa és sumar-los, però no sempre és així. Realment es poden aplicar altres operacions, com per exemple la mitjana, el mínim, el màxim o, fins i tot, el producte. L'operació que s'apliqui depèn del tipus de Mesura que agreguem i de la Dimensió al llarg de la qual ho fem. Certes operacions són incompatibles amb alguns tipus de Mesures i Dimensions.

Exemples d'operacions

La quantitat venuda durant un mes és la suma de les quantitats venudes cadascun dels dies. O per obtenir l'interès anual, multipliquem els mensuals.

L'operació d'agregació, el tipus de Mesura que agreguem i la Dimensió al llarg de la qual ho fem han de ser compatibles.

Agregació d'estocs

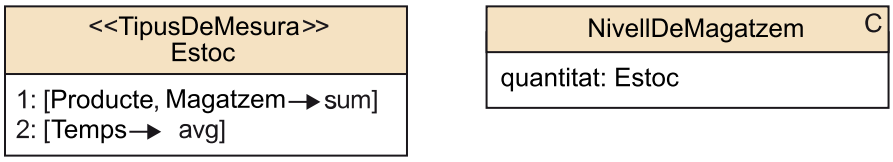
Un exemple clar d'agregació diferent segons la Dimensió són els estocs de productes als magatzems. Si registrem l'estoc diari a cada magatzem que tinguem, l'estoc mensual no serà la suma dels estocs diaris (la suma és incompatible amb els estocs i la Dimensió `Temps`). Si avui tinc un cotxe al magatzem i demà també n'hi tinc un, en total no en tinc dos, perquè realment són el mateix cotxe. El més habitual en aquest cas és fer la mitjana. En canvi, si disposo de dos magatzems i un cotxe al mateix temps a cada un d'ells, sí que tinc dos cotxes.

L'operació d'agregació més comuna és la suma. Per aquest motiu, en comptes de parlar de compatibilitat entre Mesura, Dimensió i operació, a vegades es parla simplement de Mesures additives (si es poden sumar en qualsevol Dimensió), semiadditivas (si hi ha Dimensions en les que no es poden sumar) i no additives (si no es poden sumar al llarg de cap Dimensió).

Les Mesures

Les Mesures que fan referència a ingressos acostumen a ser perfectament additives al llarg de qualsevol Dimensió. Per contra, les que fan referència a estats, com per exemple els estocs o els saldos, solen ser semiadditives (no es poden sumar al llarg del temps). Finalment, algunes que fan referència a intensitat (com per exemple, la temperatura) o es basen en fórmules matemàtiques són no additives.

Figura 19



Si no es diu el contrari, assumirem la utilització de la suma per fer *roll-up*. Si volguéssim utilitzar una operació matemàtica diferent, hauríem d'explicitar-ho com un paràmetre més de la mateixa operació (*roll-up*Nivell(cub,operació)) o en l'esquema (definint un tipus de Mesura com mostra la figura 19). En aquest cas, els estocs s'agregarien mitjançant la suma al llarg de `Producte` i `Magatzem`, i mitjançant la mitjana al llarg de `Temps`.

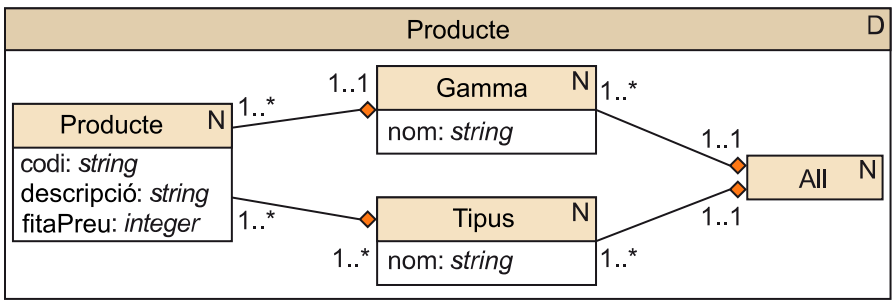
Com que certes operacions no són commutables (no és el mateix la suma de mínims que el mínim de les sumes), també podeu indicar l'ordre en el qual s'han de fer les agregacions (en l'exemple, primer es farien les sumes i després les mitjanes).

Disjuntivitat

Segons l'operació d'agregació que apliquem, correm el risc de considerar més d'una vegada la mateixa dada sense que ens n'adonem. Això passa quan hi ha un Nivell que conté instàncies amb conjunts de parts no disjunts. En aquests casos, és obligatori indicar les multiplicitats en la jerarquia d'agregació per saber quines cel·les podem utilitzar en l'agregació i quines no. En el pitjor cas, haurem d'utilitzar les instàncies de la Cel·la atòmica, que per definició han de ser disjuntes.

Certes operacions d'agregació demanen que els conjunts de parts de les instàncies que utilitzem com a operants siguin disjunts.

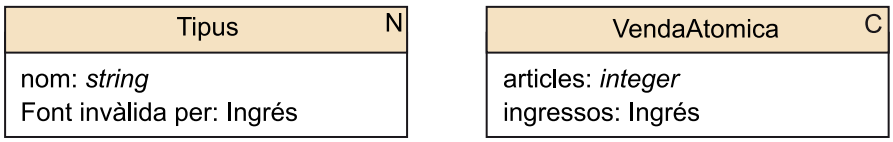
Figura 20



Exemple d'instàncies no disjuntes

Considereu ara la Dimensió `Producte` com la teniu dibuixada a la figura 20. S'ha afegit un nou Nivell: `Tipus`. Ara resulta essencial explicitar les multiplicitats de les agregacions. Observeu que l'agregació entre `Producte` i `Tipus` té multiplicitat 1..*–1..* en lloc de 1..*–1..1 com la resta de les agregacions. Això vol dir que hi ha productes que són de més d'un tipus i els conjunts de parts de les instàncies de `Tipus` no són disjunts. El producte «Kinder sorpresa» és una joguina i una xocolata. Per consegüent, pertany als tipus «Joguina» i «Xocolata» al mateix temps. Si ara intentem calcular els ingressos totals sumant els ingressos de joguines i xocolates, estarem comptant els ingressos per vendes de «Kinder sorpresa» dues vegades. Segons el tipus de mesura, això pot ser correcte o no. Podeu indicar els casos en els quals no sigui vàlid, com es mostra a la figura 21. D'aquesta manera, sabem que no hem de calcular totes les Mesures de tipus `Ingres` a partir de dades de granularitat `Tipus`, sinó que haurem de fer-ho des d'una granularitat més petita, com per exemple `Producte`.

Figura 21

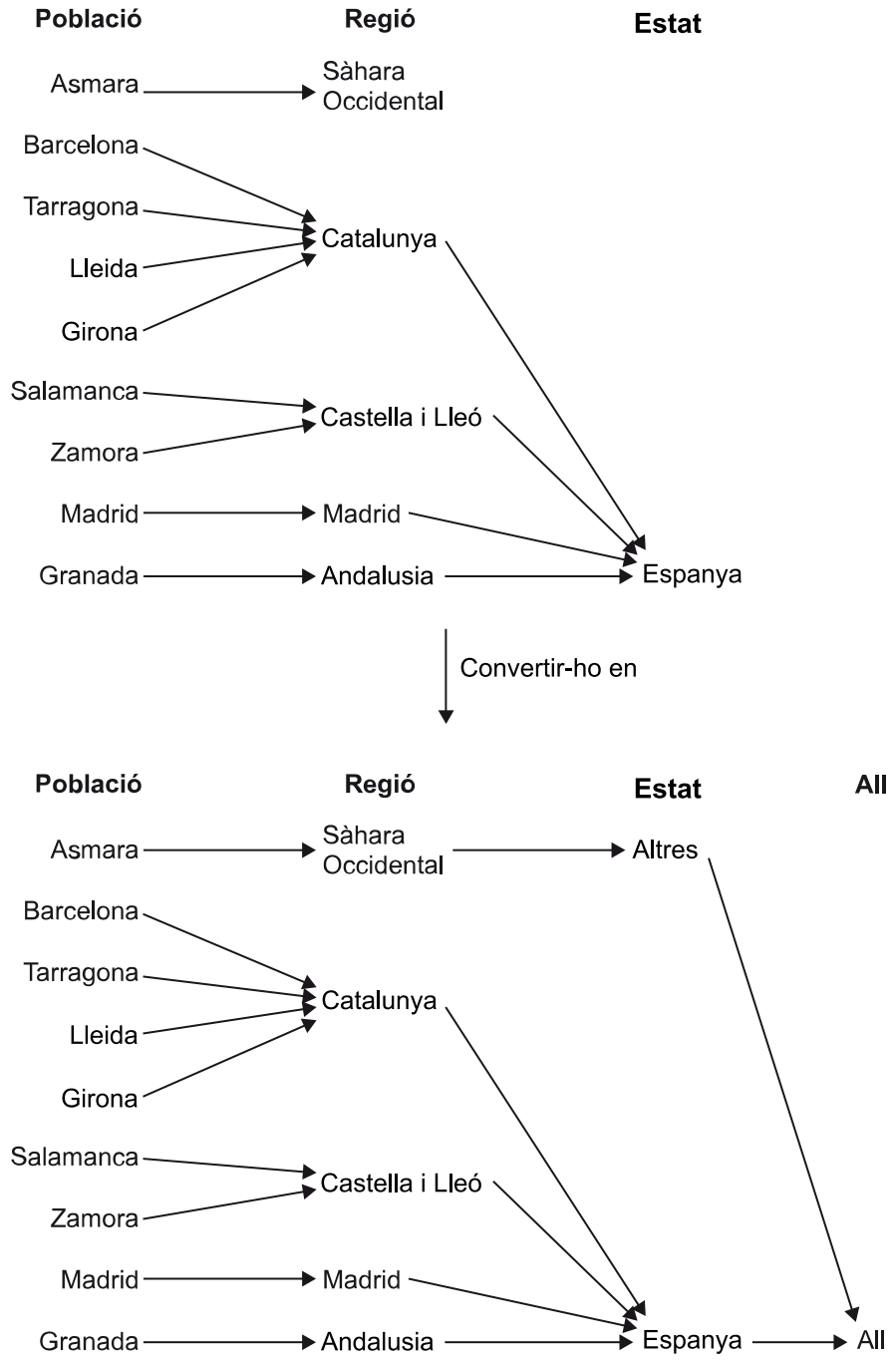


Completitut

Podria passar que alguns Nivells continguessin instàncies que no participessin en la composició de cap instància dels Nivells superiors. Això ho hem d'evitar afegint instàncies fictícies al Nivell superior, per no oblidar-nos d'operar les dades associades a aquestes instàncies. La multiplicitat mínima de l'agregació juntament amb el compost ha de ser sempre 1. Un Nivell ha de cobrir completament els Nivells que té per sota a la jerarquia d'agregació.

Hem de garantir que totes les instàncies participen com a mínim en una instància dels Nivells immediatament superiors de la jerarquia d'agregació.

Figura 22



Exemple de no completitut en la jerarquia d'agregació

Imaginem que ampliem el negoci de supermercats i n'obrim un de nou a Asmara. A més d'afegir aquesta població a la Dimensió `Lloc`, també l'afegim a la regió «Sàhara Occidental» de la que forma part. No obstant això, com que el referèndum d'autodeterminació encara no s'ha fet, decidim no registrar-li a quin Estat pertany (podeu veure com queda la Dimensió `Lloc` a la part superior de la figura 22). Com podem calcular ara el total d'ingressos? Ja no serveix consultar els ingressos obtinguts a l'Estat espanyol, perquè hi ha una regió que no és part de «Estat espanyol». Tot i que `Regio` cobreix completament `Poblacio`, `Estat` ja no cobreix `Regio`. La solució és crear una instància artificiosa d'`Estat`, que podem denominar «Altres», que tingui com a parts «Sàhara Occidental» i totes les altres regions que no formin part de cap Estat. Amb això, `Estat` ja cobreix `Regio`, però encara cal crear el Nivell `All` amb una instància que agrupi «Estat espanyol» i «Altres» (la Dimensió `Lloc` queda com està dibuixada a la part inferior de la figura 22).

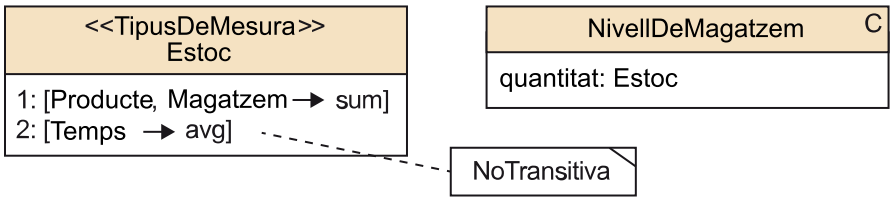
Observeu que abans d'afegir «Asmara» no feia falta el Nivell `All`, perquè el Nivell més alt de la jerarquia ja només tenia una instància.

2.3.3. Transitivitat

L'última restricció que hem de tenir en compte, també relacionada amb l'agregació, és la transitivitat de les operacions d'agregació. A vegades, no obtenim el mateix resultat si operem amb resultats parcials que si operem directament amb les dades bàsiques. Hi ha operacions que no són transitives (per exemple, la mitjana).

Hem d'assegurar-nos que les operacions d'agregació són transitives. En el cas que no ho siguin, el resultat correcte sempre és el que s'obté d'operar amb les dades atòmiques.

Figura 23



Exemple de no transitivitat

Per simplicitat, imaginem que obrim el supermercat cada dia de l'any. Cada dia del mes de gener tenim al magatzem 100 unitats, cada dia del mes de febrer 300 i cada dia del mes de març 200. Sembla clar que la mitjana d'unitats que tenim al gener és 100, al febrer 300 i al març 200. Quin és l'estoc durant el primer trimestre? 200 unitats? No podem fer la mitjana de les mitjanes mensuals. Hem de fer la mitjana dels estocs diaris. Realment, la mitjana és 196,6 (100 × 31 + 300 × 28 + 200 × 31 = 196,6). Si interessa reflectir això, ho podem fer com es pot veure a la figura 23.

3. Disseny conceptual

Tot i que les eines OLAP es consideren relativament fàcils d'utilitzar, construir-les i mantenir-les requereix coneixements especialitzats. Per aquest motiu, dedicarem bona part d'aquest mòdul a estudiar-ne el disseny. De la mateixa manera que per a les bases de dades operacionals, ho farem en tres passos: disseny conceptual, disseny lògic i disseny físic. En aquest apartat veurem com es pot fer un disseny conceptual multidimensional.

Figura 24

El primer que farem és aconseguir un esquema expressat amb UML dels cubs de dades que ens interessen.

Un Fet i el seu corresponent conjunt de Dimensions formen una Estrella.

Un conjunt d'Estrelles formen una Constel·lació.

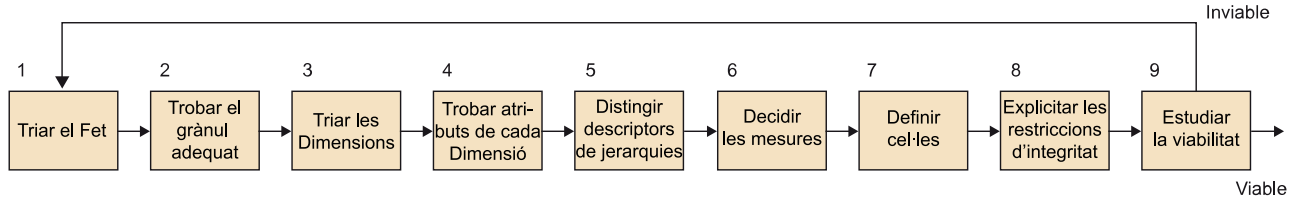
Podem representar una Estrella com un paquet (com podeu veure a la figura 24). Vegem ara com podem dissenyar el que hi ha dins de cadascun d'aquests paquets.

3.1. Una metodologia per dissenyar una Estrella

Una dels avantatges del disseny multidimensional és que utilitza la tècnica de «divideix i venceràs». No intentem dissenyar de cop tots els magatzems departamentals de la nostra empresa, ni satisfer al mateix temps les necessitats dels analistes, sinó que fem petits dissenys bastant independents els uns dels altres.

Dissenyem per separat cada Estrella i ho fem en un procés iteratiu que consta de nou passos.

Figura 25



3.1.1. Triar el Fet

Primer de tot, hem de saber quin tema estudiarem amb la nostra Estrella: quin serà el Fet. Un Fet no és més que un conjunt d'esdeveniments amb dades numèriques associades. Els candidats habituals són els processos de negoci. Per **processos de negoci** entenem tots els processos operacionals de l'organització que estan suportats per algun sistema informàtic del qual es poden extreure dades. Probablement, els mateixos analistes us suggeriran el procés de negoci que heu de modelar. Això no obstant, us podeu adonar del condicionament tan fort que tenim per a l'elecció d'un Fet: hem de disposar de les dades, preferiblement en suport informàtic.

El primer que hem de fer per dissenyar una Estrella és triar el Fet objecte d'anàlisi.

En primer lloc, intenteu definir Fets «petits». Definir de cop una única Estrella que inclogui tots els àmbits de l'empresa és pràcticament impossible. El més fàcil és començar desenvolupant Estrelles per analitzar processos de negoci que estiguin recolzats per un únic sistema operacional. Per exemple, si tenim un sistema de gestió de vendes, ens resultarà relativament fàcil dissenyar i posar en funcionament una Estrella per a l'anàlisi de vendes.

3.1.2. Trobar el gránul oportú

El gránul és l'individu últim que volem analitzar, la Cel·la més petita que volem tenir disponible.

Ja no parlem d'un concepte gran i abstracte com un procés de negoci, sinó del tipus d'objecte concret que volem analitzar. Els gránuls més típics són les transaccions individuals (per exemple, les vendes) o les instantànies diàries d'un estat (per exemple, els estocs). Generalment, sol ser qualsevol tipus d'esdeveniment que es produeixi amb una freqüència relativament alta.

Exemple de gránuls possibles

Els gránuls candidats a la nostra Estrella de vendes podrien ser «vendes mensuals de producte per botiga», «vendes diàries d'un cert producte a un client en una població», «compres que fa un client en un moment concret en una determinada botiga» o, fins i tot,

«línies dels tiquets de compra», que representen la venda d'un producte en un moment concret en una determinada botiga.

Triar el grànul del nostre Fet és molt important, perquè determina la dimensionalitat de la base de dades i, com veurem quan estudiem la viabilitat de la implementació, té un impacte directe en la mida del conjunt de dades. Un grànul molt petit resultarà en una base de dades molt gran. Per contra, un grànul més gran implicarà renunciar a alguna Dimensió o, més possiblement, a calcular certes Mesures, a causa, per exemple, de problemes amb la transitivitat de les operacions o la no disjuntivitat dels compostos.

Mida d'un grànul

Entenem que un grànul és més gran com més esdeveniments representa.

D'aquesta manera, és més gran la granularitat `Mes` que `Dia` i `Dia` que `Hora`.

Exemple d'elecció del grànul

Què passa si triem com a grànul «vendes mensuals de producte per botiga»? Com sabrem quin dia del mes hem venut més? Aquest seria un grànul massa gran. Ara bé, penseu els problemes que pot generar triar com a grànul «línies de bitllets de compra». Cada vegada que una caixera passa un article pel lector de codis de barres, generem una cel·la al nostre sistema. Quant triga una caixera a fer-ho? Quantes caixeres tenim? Quant temps treballen a diari? Probablement, el sistema acabarà contenint més dades de les que pot gestionar. Si realment no ens fa falta tant detall, hauríem d'intentar reduir el volum de dades d'alguna manera. En el nostre exemple, pensem que els sistemes operacionals identifiquen a cada client (per exemple, amb el número de targeta de crèdit) i que un mateix client compra el mateix article moltes vegades, de manera que podem generar una única cel·la per a totes les seves compres d'un mateix producte, en tot un dia, dins de la mateixa població.

Contingut complementari

Vegeu també

Ja hem vist aquests conceptes a l'apartat «Restriccions d'integritat inherents al model».

Un bon disseny sempre demana triar el grànul més petit possible, no perquè els usuaris vulguin veure sempre les dades més detallades, sinó per no perdre la possibilitat de calcular cap dada derivada amb el mínim error. El límit sempre és la disponibilitat de dades en els sistemes operacionals. No podem triar «línies dels tiquets de compra», si el sistema operacional corresponent només registra la quantitat total de la compra.

Triar un grànul massa gran representa perdre informació. No obstant això, triar-lo **massa** petit pot representar malbaratar espai o arribar a fer inviable el projecte per excés de dades.

3.1.3. Triar les Dimensions que s'utilitzaran en l'anàlisi

Unes Dimensions típiques són `Temps`, `Producte`, `Client`, `Promocio`, `Magatzem`, `Tipus` o `Estat`. Si el grànul està clarament definit, trobar un primer conjunt de Dimensions d'anàlisis és immediat a partir de la mateixa definició. A aquest primer conjunt inicial li podem afegir altres Dimensions, però només si cada combinació de les instàncies de les Dimensions inicials determina una instància de la Dimensió que volem afegir. Si en determinés més d'una, significaria que hem de reconsiderar la Dimensió o el grànul mateix per poder-la afegir.

La multiplicitat de les associacions entre Fets i Dimensions sempre és molts a un. Moltes instàncies del Fet s'associen amb la mateixa instància de Dimensió, però només una instància de Dimensió s'associa amb cada instància del Fet.

Exemple d'elecció de Dimensions

Si el nostre grànul és «vendes diàries d'un cert producte a un client en una població», aleshores tindrem almenys les Dimensions `Temps`, `Producte`, `Client` i `Lloc`. A més d'aquestes quatre Dimensions, n'hi ha d'altres, com per exemple `Promocio`, que queden determinades per la resta. Un determinat producte en un moment donat i en una població està essent promocionat d'alguna manera (per exemple, amb un 10 % de descompte) i volem estudiar com afecta això a les vendes. Per contra, no podríem afegir la Dimensió `Publicitat` perquè no podem associar cap tipus de publicitat en concret a una cel·la. Imaginem que hi ha cartells publicitaris a les carreteres, que coincideixen amb diferents campanyes radiofòniques locals i altres campanyes televisives estatals. Com podem associar això a una venda en una població? Si el que s'anuncia és un establiment, com l'associarem a la venda d'un producte? Les compres poden no coincidir exactament en el temps amb les campanyes publicitàries que les provoquen. Com associarem una campanya a un dia concret? `Publicitat` no queda determinada per `Temps`, `Producte`, `Client` i `Lloc`.

El grànul mateix ja determina un primer conjunt de Dimensions. Hi hem d'afegir els altres punts de vista que vulguem utilitzar en l'anàlisi i que quedin determinats pel conjunt inicial de Dimensions.

El més habitual és trobar entre quatre i quinze Dimensions. Trobar-ne dues o tres és molt estrany i us hauria de fer sospitar que hi ha alguna Dimensió més que es podria afegir al disseny (per exemple, la Dimensió temporal és gairebé omnipresent). En l'extrem oposat, que n'hi hagi vint tampoc no sembla fàcilment justificable. Moltes d'aquestes Dimensions resultaran prescindibles o en trobarem algunes parelles o grups que estaran fortament correlacionats, de manera que es podrien representar com una única Dimensió, com veurem a l'apartat «Dependències entre Dimensions».

Les Dimensions

Heu d'entendre les Dimensions com les variables independents que afecten cada observació. Penseu que resulta difícil trobar un fenomen amb moltes variables i un amb poques no és massa interessant d'analitzar.

3.1.4. Trobar els atributs de cada Dimensió

D'entre els atributs que podem trobar en els sistemes operacionals, hem de triar els que pertanyen a les Dimensions i ens seran útils per triar i descriure l'espai d'anàlisi. Hem de seleccionar qualsevol atribut que creguem que pugui ser útil per seleccionar, agrupar o simplement posar com a capçalera d'un informe. Una Dimensió pot tenir amb relativa facilitat més de cinquanta atributs. Cal documentar tant l'origen com la interpretació de cadascun dels atributs.

Com que els atributs de Dimensió han de servir per fer seleccions i agrupacions, han de tenir sempre un domini discret, mai continu. Com seleccionareu un valor, si té infinits decimals? A més, un atribut mai no ha d'estar codificat ni abreujat i ha de ser fàcilment intel·ligible per a l'usuari (en principi, un codi de barres no és un bon atribut per a la Dimensió `Producte`). Pot interessar tenir alguns codis no directament interpretables (com per exemple el codi de barres) per identificar les instàncies dins del mateix sistema. No obstant això, com que mai no seran utilitzats per l'usuari, hem de calcular molt bé l'espai que podem estalviar-nos. Finalment, també cal dir que la qualitat d'un atribut ha d'estar garantida: no hi poden haver errors en l'escriptura.

Exemple de Dimensió temporal

Per exemple, en una Dimensió temporal, a més de saber el dia de la setmana, el mes o l'any, tindríem que poder seleccionar dies festius o laborables, setmanes de vacances o de classe, de matrícula o d'exàmens, anys normals o de traspàs, etc.

Quan es van definir les Dimensions dins de l'apartat «Estructures de dades», en «Components del model multidimensional», ja es va parlar d'atributs de Dimensió i de les característiques més importants. No obstant això, a tall de recordatori, tenim la definició següent:

Un atribut de Dimensió ha d'estar definit sobre un domini discret, i ha de ser descriptiu, fàcil de recordar i comprensible a primera vista. Els atributs textuals solen pertànyer a les Dimensions, mentre que els atributs numèrics són habitualment Mesures.

Exemple d'atribut de Dimensió

Podrem considerar que el preu d'un producte (un atribut numèric) ens ve donat i que mai no canviarà. Si creiem que ens pot ser útil per seleccionar vendes, estaríem parlant d'un atribut de la Dimensió `Producte` en l'Estrella `Vendes`.

Si un atribut ens sembla interessant i no compleix les característiques desitjables, cal que el modifiquem fins que prengui la forma adequada: fent-lo més explicatiu (per exemple, canviant codis per frases), definint rangs de valors (per exemple, preus de menys d'un euro, d'entre un i cinc euros, d'entre cinc i cinquanta o de més de cinquanta), netejant-lo (per exemple, eliminant valors nuls o arreglant errors d'ortografia), etc.

S'ha de convertir qualsevol tipus de codi que hi hagi en els sistemes operacionals en un text explicatiu i discretitzar els dominis continus per facilitar la selecció i comprensió dels atributs de Dimensió.

3.1.5. Distingir entre descriptors i jerarquies d'agregació

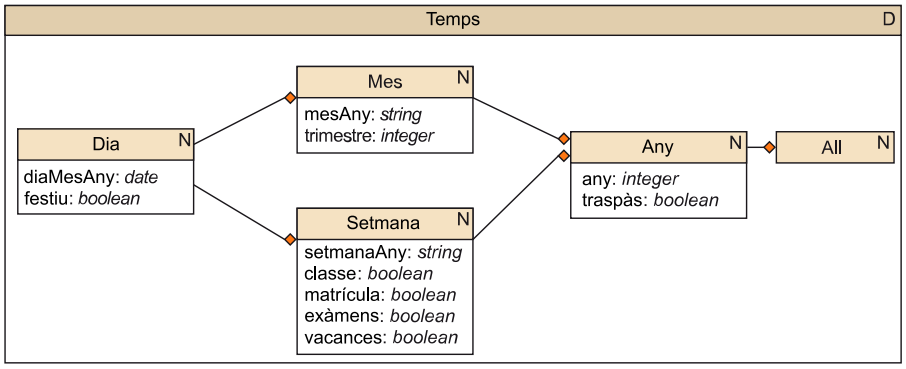
D'entre els atributs que hi ha en una Dimensió, hem de distingir-ne dos tipus: els que utilitzarem per agrupar i els que serviran simplement per seleccionar. Per exemple, utilitzarem els mesos i els anys per agrupar dies, mentre que

usarem la festivitat o no festivitat d'un dia simplement per seleccionar. Els primers són els que defineixen les jerarquies d'agregació, i els segons són els descriptors.

Un atribut per si mateix no és d'un tipus o d'un altre. Que un atribut defineixi un Nivell dins d'una jerarquia d'agregació o no depèn molt més del que els usuaris desitgen, que de les dades en si. Les jerarquies s'haurien de consensuar amb els usuaris. Amb jerarquies massa grans augmentem la complexitat del sistema, però amb jerarquies massa simples podem perdre possibilitats d'anàlisi o retardar el sistema.

El Nivell atòmic de les jerarquies queda definit pel grànul que hàgim triat. Si tenim vendes diàries, el Nivell atòmic de la Dimensió `Temps` serà `Dia` i no `Hora` ni `Mes`. A partir d'aquest Nivell, la resta del graf està definit per les dependències funcionals que hi hagi entre els atributs d'agrupació que triem. Col·locarem la resta dels atributs com a Descriptors en el Nivell que els correspongui.

Figura 26



Una altra interpretació de la jerarquia d'agregació

Si tornem a observar ara la Dimensió de la figura 8, podem veure que `Dia` determina `Setmana` i `Mes`, però ni `Setmana` determina `Mes` ni `Mes` determina `Setmana`, encara que tots dos determinen `Any`. Recordeu que les dependències funcionals compleixen la propietat transitiva. Per consegüent, com que `Mes` determina funcionalment `Trimestre` i aquest determina `Any`, podem dir que `Mes` determina `Any`. No obstant això, recordeu també que les agregacions compleixen igualment la propietat transitiva i que no cal explicitar la que hi ha entre `Mes` i `Any`. Si els usuaris no utilitzessin els trimestres per agrupar mesos, aquest Nivell desapareixeria de la jerarquia (seria simplement un atribut associat a `Mes`, com podeu veure a la figura 26), i llavors sí que dibuixaríem l'agregació entre `Mes` i `Any`. En aquesta figura, hem dibuixat a cada Nivell sol els Descriptors que en depenen directament.

Podeu entendre les agregacions que formen una jerarquia d'agregació com a dependències funcionals entre atributs de Dimensió que utilitzem per agrupar.

3.1.6. Decidir quines són les mesures que interessen

De manera típica, les Mesures són atributs numèrics que es poden sumar, com per exemple les quantitats venudes o els ingressos que genera una venda. Podeu triar qualsevol atribut que tingueu directament disponible a les bases de dades operacionals o que pugueu derivar a partir dels que tingueu. Penseu només que l'espai que ocupin les Mesures (és a dir, la mida de les cel·les) afectarà sensiblement al volum total de la base de dades (si ocupen molt, la base de dades serà massa gran per ser funcional). El millor és trobar totes les Mesures interessants per als analistes i després triar només les que pensem que seran més útils.

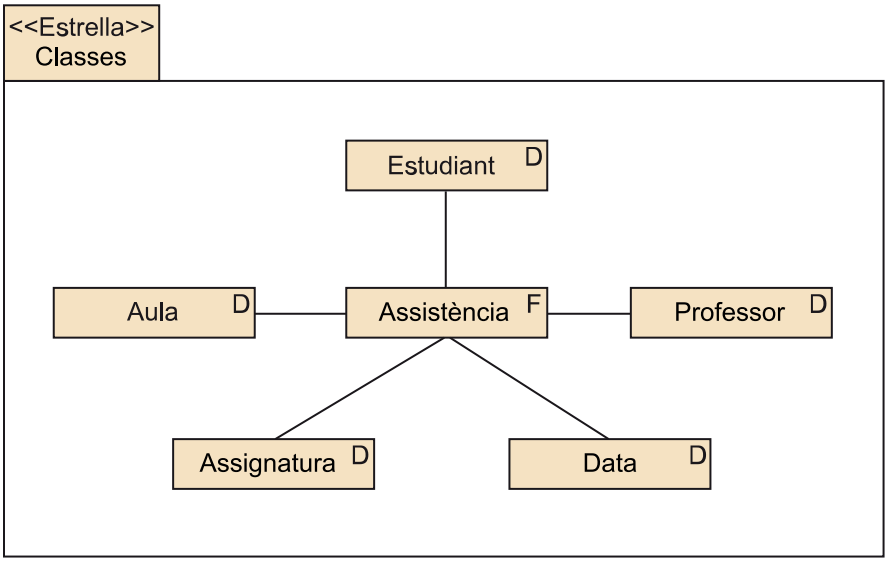
Nota

Si una Mesura és derivada, cal que ho indiqueu com faríeu amb qualsevol atribut amb UML (posant una barra al davant del nom i adjuntant un comentari amb la fórmula utilitzada en la derivació).

Les Mesures són atributs numèrics normalment additius.

Generalment, com més Mesures contingui un Fet, més útil serà l'Estrella. No obstant això, a vegades trobareu alguns Fets molt útils que no tenen cap Mesura. Com ja hem dit abans, les instàncies dels Fets poden representar esdeveniments. Normalment, en donar-se els esdeveniments, mesurem alguna cosa. No obstant això, hi ha casos en els quals únicament volem tenir constància que l'esdeveniment ha tingut lloc. En aquest cas, el Fet no té cap Mesura.

Figura 27



Exemple de Fet sense Mesures

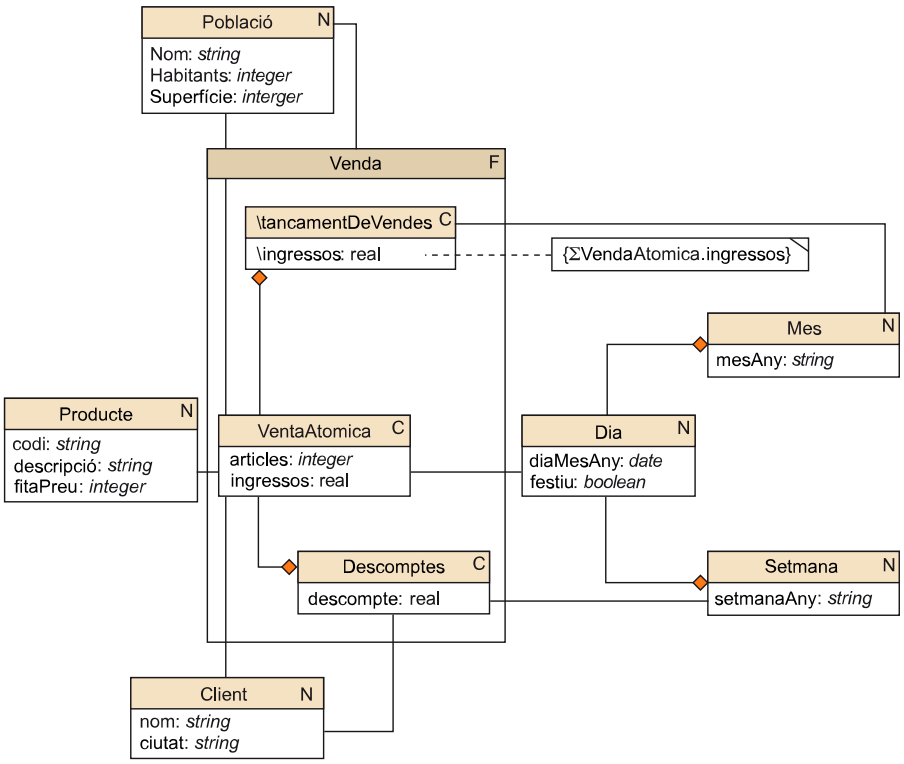
Penseu que estem en una universitat presencial i volem analitzar l'assistència a classe de cada alumne, segons l'assignatura, el professor que imparteix la classe i l'aula i la data en els quals es fa. Tindríem l'Estrella de la figura 27. `Assistencia` no tindria cap Mesura, perquè l'única cosa que ens interessa és tenir constància de si un alumne ha assistit a classe o no. Amb aquesta Estrella sense Mesures, ja podem respondre consultes, com per exemple quin professor té més alumnes a classe o quants professors han fet alguna classe en una determinada aula.

Hi ha alguns Fets que no tenen cap Mesura.

3.1.7. Definir Cel·les

Veureu que en el conjunt de Mesures que heu triat hi ha diferents granularitats. Cada una pertany a una de les Cel·les del Fet i heu d'anar amb molta cura per posar-les allà on els correspon. Heu de dibuixar Cel·les de diferent granularitat dins del mateix Fet, per contenir les Mesures de la granularitat corresponent. De tot el reticle de Cel·les que teniu exemplificat a la figura 14, només cal que dibuixeu les Cel·les que contenen Mesures o que considereu especialment rellevants.

Figura 28



Exemple de Mesures de diferent granularitat

Pensem que ens interessen quatre Mesures: el nombre d'articles venuts, els ingressos que genera una venda, els ingressos que tenim mensualment a cada població (quan tanquem els comptes) i el descompte que fem a cada client. Considerem que el descompte s'aplica de manera setmanal i depèn de la despesa que ha fet aquest client durant el període de temps corresponent i del tracte que hi tinguem. A la figura 28, podeu veure com quedaria l'esquema. Apareixen tres Cel·les diferents que representen les tres granularitats que ens interessen. Cada Cel·la conté les Mesures que tenim en aquesta granularitat. Observeu que els ingressos mensuals per població són una Mesura derivada que s'obté sumant els ingressos que tenim en Cel·les de granularitat més petita. Com que es tracta de l'única Mesura que conté aquesta Cel·la, també podem marcar la Cel·la com a derivada (amb una barra davant del nom). Per contra, no podem calcular el descompte directament de les altres Mesures, perquè depèn de l'oferta que fem a cada client en cada moment. Per simplificar la representació, podem considerar que, si una Cel·la no està associada a cap Nivell d'una Dimensió, ho està al Nivell `All`. D'aquesta manera, encara que no quedi reflectit en el dibuix, Descomptes està associat al Nivell `Poblacio::All` i al Nivell `Producte::All`.

De totes les Cel·les que expliciteu, caldrà emmagatzemar-ne algunes de manera física i unes altres només contindran Mesures que es podran obtenir simplement fent *roll-up* i aplicant l'operació que correspongui (i que marqués- mos com a derivades). No obstant això, en el disseny conceptual no hem de considerar si unes dades derivades s'emmagatzemen físicament o simplement es calculen quan és necessari. Això ja serà una decisió de disseny físic. En aquest moment, només cal deixar constància de tot el que es consideri especialment rellevant.

Cal explicitar les Cel·les que contenen Mesures interessants per als analistes, encara que siguin derivades. Podríem no posar tots els elements que siguin derivats a l'esquema. Els explicitem només per remarcar com són d'importants.

3.1.8. Explicitar les restriccions d'integritat

Una vegada ja tenim totes les Mesures, Cel·les i Nivells, només ens falta expressar les restriccions d'integritat corresponents (Bases, restriccions d'agregació i restriccions de transitivitat), com ja hem vist a l'apartat «Restriccions d'integritat inherents al model».

Com a últim pas del disseny conceptual, cal expressar les restriccions d'integritat.

Com especialment importants, en aquest moment cal destacar les Bases. El conjunt inicial de Dimensions que trobem després de definir el grànul ha de donar lloc a una Base. Si hi substituïm Dimensions segons les dependències funcionals que hi ha amb la resta de les Dimensions, podem obtenir les altres Bases de l'espai.

Exemple de Bases d'un espai

Ja hem vist abans que una base de `VendaAtomica` és {`Producte, Dia, Client, Poblacio`}. En el cas d'afegir la Dimensió `Promocio`, aquesta no formaria part de la Base, perquè queda completament determinada per les Dimensions `Temps, Lloc` i `Producte` (en un moment concret, en una determinada població, un producte sol es promociona d'una manera). Si pensem que una promoció determina un producte i que en tot moment tots els productes tenen una promoció o una altra (també podem considerar la no promoció com una mena de promoció), llavors podem substituir `Producte` per `Promocion` i obtenir una segona Base de l'espai {`Promocio, Dia, Client, Població`}. La Base de `Descomptes` seria {`Client, Setmana`} i la de `TancamentDeVendes` seria {`Poblacio, Mes`}.

3.1.9. Estudiar la viabilitat

Una vegada ja hem acabat el disseny conceptual, cal veure si l'Estrella és realment implementable o no. S'ha d'estimar l'espai que ocuparan totes les dades. La manera de saber quin espai ocuparan consisteix a mirar el contingut dels

sistemes operacionals. Si això és massa complicat, podem fer una estimació. Per fer-ho, podem considerar només el que ocuparà emmagatzemar les instàncies del Fet. El volum de dades que puguin ocupar les Dimensions resulta en general insignificant respecte al que ocuparà el Fet.

Per fer l'estimació del que serà necessari per emmagatzemar totes les instàncies d'una Cel·la, calculem la grandària de l'espai que defineix cadascuna de les seves Bases, i multipliquem el nombre d'instàncies de cada Nivell que les formen. L'espai més petit ens donarà el nombre màxim d'instàncies que pot tenir la Cel·la.

Exemple d'estimació d'instàncies

Pensem que tenim 1.000 clients i 1.000 productes diferents, venem a 10 poblacions i volem emmagatzemar dades durant 1.000 dies. A més, tenim una mitjana de tres promocions diferents per a cada producte (31.000 promocions en total). Si prenem la Base {`Producte, Dia, Client, Poblacio`}, obtenim un espai de 1.000 × 1.000 × 1.000 × 10 (10^{10} possibles cel·les). Per contra, si considerem la Base {`Promocio, Dia, Client, Poblacio`}, obtenim un espai de 3.000 × 1.000 × 1.000 × 10 ($3 \cdot 10^{10}$ possibles cel·les). D'aquesta manera, el nombre màxim de cel·les que podem tenir en el nostre espai és 10^{10}.

Aquesta fita pot resultar una mica excessiva. S'ha d'intentar afinar més, sabent com queden de disperses en aquest espai les cel·les que volem analitzar-hi.

Exemple de millora en la fita

Un client comprarà a totes les nostres botigues? La resposta, probablement, és que no. Comprarà només a les que tingui més a prop de casa. No es donaran totes les combinacions client-població i, si es donessin, no passaria cada dia. Podem considerar que un client només comprarà en una població. D'aquesta manera, encara que l'espai sigui de 1.000 × 1.000 × 1.000 × 10, sabem que tindrem 1.000 × 1.000 × 1.000 cel·les com a molt. Si a més estimem que la mitjana d'articles diferents que compra un client al dia és 10, ens queden aproximadament 10 × 1000 × 1000 cel·les.

Un cop sabem quantes cel·les tindrem, multipliquem aquesta xifra pel nombre de *bytes* que ocuparà cada cel·la (6 *bytes* de les Mesures articles i ingressos més 20 *bytes* dels identificadors de les 5 Dimensions, en el nostre cas) i obtindrem una estimació del que ocuparà la nostra Cel·la ($26 \times 10^7 bytes = 26 \times 10^4 kbytes = 260 \times 10^1 Mbytes = 2{,}6\ Gbytes$). Generalment, el que ocupa la Cel·la atòmica són ordres de major magnitud que el que ocupen les altres Cel·les (`Descomptes` ocuparia aproximadament 1,7 *Mbytes* i `TancamentDeVendes`, només 4,3 *kbytes*). Només si tenim Dimensions molt petites o moltes Cel·les en el nostre esquema, realment cal sumar aquestes quantitats.

Per saber quant espai ocuparà la nostra Estrella, el més realista és observar les dades que contenen els sistemes operacionals (les nostres fonts de dades) per saber quantes n'hi haurà en el nostre sistema d'anàlisi. Si no és possible, també ho podem estimar a partir del nombre d'instàncies del Nivell atòmic de cada Dimensió i la mida de la Cel·la.

Serà capaç de gestionar aquest volum de dades el nostre sistema? Serà prou bona la velocitat de resposta? Podem carregar dins de la finestra d'actualització totes les dades necessàries per mantenir-ho actualitzat? Si la resposta a qualsevol d'aquestes preguntes és que no, només tenim dues opcions: 1) millorar el programari i el maquinari que tenim o 2) replantejar el disseny triant un gràNul no tan fi o descartant algunes Mesures que no siguin essencials.

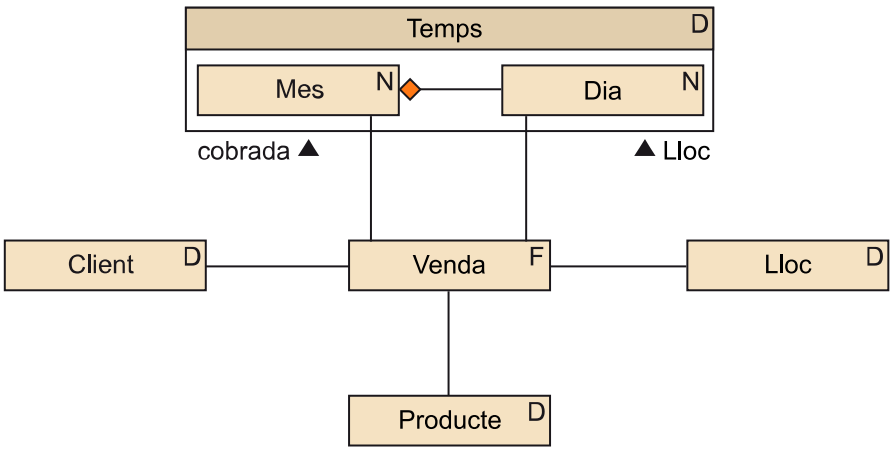
3.2. Reconsideracions en el disseny conceptual

En aquest apartat veurem algunes millores que, en alguns casos, es poden fer en el disseny d'una Estrella.

3.2.1. Dimensions amb múltiples rols

Fins ara, totes les Dimensions que hem utilitzat per analitzar un Fet eren molt diferents. No obstant això, és bastant habitual que hi hagi Dimensions tan semblants que, de fet, hauríem de parlar de la mateixa Dimensió repetida, que té rols diferents. A vegades, d'una Dimensió a una altra, només canvia la jerarquia d'agregació que té definida o els Descriptors que ens resulten interessants. Normalment, ens quedarem amb una sola Dimensió i l'associarem amb el Fet tantes vegades com sigui necessari. No obstant això, en alguns casos –per exemple, per motius de confidencialitat o limitacions en l'eina OLAP–, podria interessar mantenir les Dimensions diferenciades.

Figura 29



Exemple de Dimensió amb rols múltiples

Si en el nostre sistema de vendes admetem el pagament endarrerit (a trenta, seixanta i noranta dies, per exemple), tindrem dues dates associades a cada venda, primer aquella en la qual s'ha produït la venda realment i després aquella en la qual hem cobrat. Això podem representar-ho com veieu a la figura 29. Podria passar que les dues associacions fossin tant al mateix Nivell com a Nivells diferents. Si l'única cosa que ens interessés fos el mes que hem cobrat, tindrem que el final de l'associació `cobrada` es relaciona amb el Nivell Més en lloc de Dia, però la Dimensió continua essent la mateixa.

Una Dimensió pot estar associada més d'una vegada amb un mateix Fet i tenir rols diferents.

3.2.2. Dependències entre Dimensions

El quocient entre el nombre màxim teòric de cel·les que pot arribar a haver-hi i les cel·les que hi ha realment és la dispersió del cub.

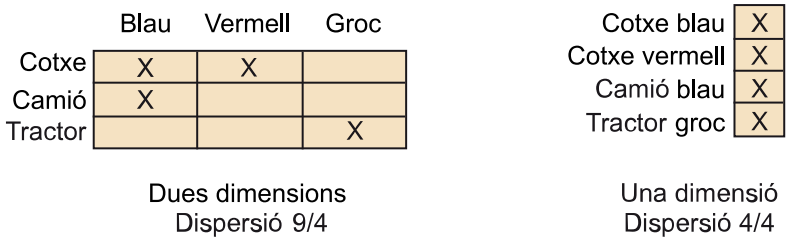
Una dispersió molt gran pot generar problemes en la gestió dels cubs. La causa de la dispersió és que certes combinacions d'instàncies de les Dimensions no són vàlides. El possible valor d'una de les Dimensions depèn dels valors de les altres. Com més dependguin unes Dimensions de les altres, més alta serà la dispersió.

Si el valor d'una Dimensió depèn del valor d'una altra, diem a estan correlacionades.

Quan veiem que dues Dimensions estan correlacionades, les podem ajuntar per obtenir una única Dimensió. Aquesta nova Dimensió tindrà una instancia per a cada element del producte cartesià de les Dimensions originals que realment tingui sentit. Tot i que som capaços de representar la mateixa informació, fent-ho reduïm el nombre de Dimensions, però com a contrapartida les que queden són més grans.

La correlació entre dues Dimensions no ve d'aquestes, sinó del conjunt de dades que analitzem. En una Estrella podem veure que dues Dimensions estan molt correlacionades, mentre que en una altra no ho estan en absolut.

Figura 30



Exemple de Dimensions correlacionades

Imaginem que parlem d'un concessionari de vehicles, que està interessat a analitzar les vendes segons el tipus de vehicle i els colors. Aquest concessionari és molt particular, ja que, com teniu esquematitzat a la figura 30, només ven cotxes vermells o blaus, camions blaus i tractors grocs. El color depèn molt del tipus de vehicle. Per tant, si les seves Dimensions d'anàlisi són Vehicle i Color, té un espai de mida nova que només contindrà quatre cel·les. Per contra, si tinguéssim una única Dimensió VehicleAcolorit, que només té instàncies per a les parelles vehicle-color que realment hi ha, la dispersió seria mínima

(hi hauria una cel·la en cada punt de l'espai). Abans teníem dues Dimensions de mida tres i ara només en tenim una, però de mida quatre.

Hem de fusionar Dimensions només quan estiguin realment correlacionades i ens representi un guany en la dispersió del cub sense que el nombre d'instàncies de les Dimensions creixi de manera excessiva. Penseu que els usuaris utilitzaran les Dimensions per seleccionar les dades que volen veure. Com més instàncies tinguin les Dimensions i més complexos siguin els conceptes que representen, més difícil els resultarà fer seleccions.

Exemple de Dimensions que no interessa fusionar

A la nostra Estrella de vendes, hem vist que molt probablement un client sempre compraria a la població on resideix i que això generava certa dispersió en el cub. No obstant això, si ajuntem les Dimensions `Client` i `Poblacio`, no obtenim cap concepte concret. Cap analista no voldrà consultar parelles client-població. Voldrà consultar les vendes a un client i les vendes en una població, majoritàriament per separat. A més, cap client no té prohibit comprar en una població. Per consegüent, si les poblacions són prou pròximes, és possible que l'espai `Client × Poblacio` no sigui gaire dispers i que qualsevol client hagi comprat alguna vegada a cada una de les poblacions. La Dimensió `ClientPoblacio` tindrà una instància per gairebé cada element del producte cartesià entre `Client` i `Poblacio`.

Si volem reduir la dispersió, el que hem de fer és fusionar les Dimensions que estiguin molt correlacionades. Això augmentarà la mida de les Dimensions, però reduirà la dispersió del cub.

3.2.3. Minidimensions

En alguns casos, podem observar que una Dimensió és massa gran. D'una banda, això pot provocar dispersió al cub, tret que el Fet també tingui moltes instàncies; i de l'altra, dificulta les consultes. Per solucionar-ho, podem crear una Dimensió més petita només amb els atributs més utilitzats. Sovint, amb això no n'hi ha prou i el que hem de fer és, a més, modificar el domini d'algun d'aquests atributs perquè l'atribut tingui menys valors possibles.

Tenir aquesta minidimensió no implica eliminar la Dimensió original. Poden coexistir totes dues dins de la mateixa Estrella, però han d'aparèixer vinculades per una associació.

Exemple de minidimensió demogràfica

Algunes empreses tenen milions de clients (penseu, per exemple, en telefonia). En aquests casos, la Dimensió Client és excessivament gran i dificulta de manera clara les consultes. Per solucionar aquest problema, es defineixen perfils de clients. Es prenen només els atributs més utilitzats a les consultes (per exemple, edat, sexe, estat civil, nivell adquisitiu, etc.) i amb ells es crea una Dimensió demogràfica. Si es manté la Dimensió Client juntament amb la demogràfica, s'ha de definir una associació entre aquestes que vinculi cada client amb el seu perfil. Si aquesta Dimensió demogràfica encara té massa instàncies, podem crear rangs de valors més petits en algun dels atributs. Per exemple, podem parlar només de tres valors de l'atribut edat: edats entre 0 i 30, 31 i 60 i més de 60.

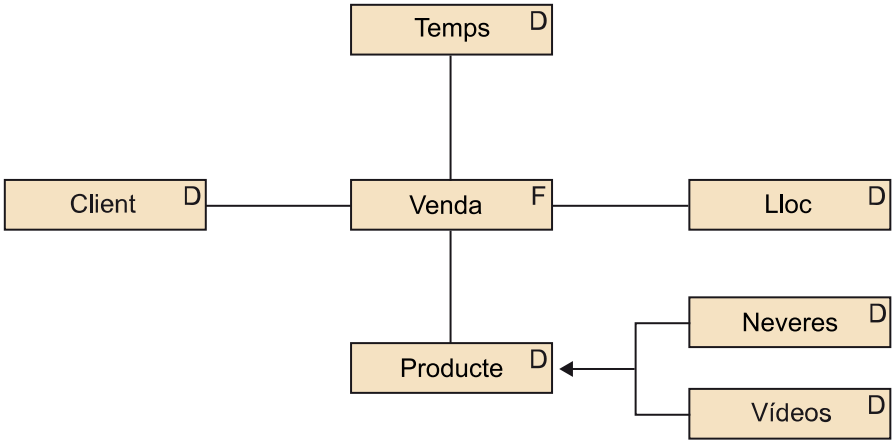
Quan tenim Dimensions massa grans, podem crear minidimensions amb els atributs més utilitzats per facilitar les consultes dels usuaris.

Observeu que el nombre d'instàncies d'una minidimensió (com per exemple la demogràfica) no depèn del nombre d'instàncies de la Dimensió (per exemple, `Clients`), sinó del nombre d'atributs que té i la mida dels seus dominis. Si veiem que disposem de molts atributs per fer una minidimensió, podem fer-ne més d'una. Si portem això a l'extrem, podem fer una minidimensió per a cada atribut de la Dimensió original.

3.2.4. Disseny de dades heterogènies

Les instàncies d'algunes Dimensions (com per exemple Productes) no són gaire homogènies. Els atributs que valen per a un producte no tenen sentit per a un altre (per exemple, les neveres tenen litres de capacitat, mentre que els aparells de vídeo disposen de sistema d'enregistrament). Això es tradueix en l'aparició de molts valors nuls (que dificulten les consultes) i la impossibilitat de definir certes jerarquies d'agregació perquè no són vàlides per a totes les instàncies.

Figura 31

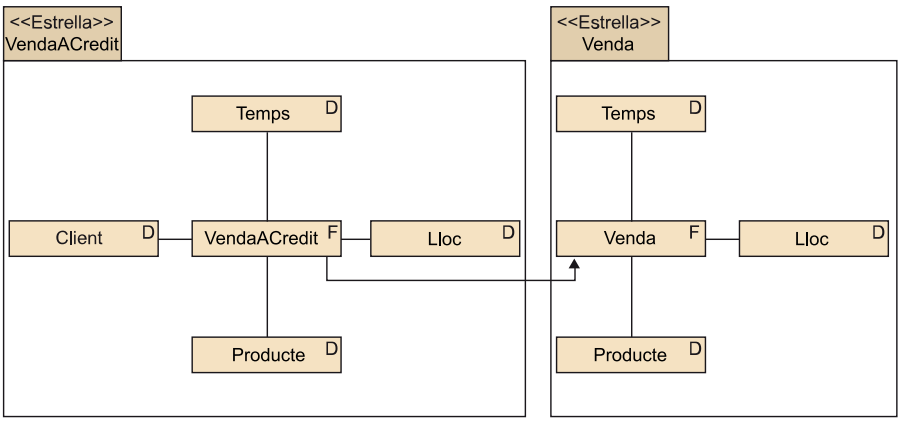


La manera de solucionar aquest problema consisteix a especialitzar les Dimensions (com podeu veure dibuixat a la figura 31). D'una banda, tenim la Dimensió general que té els Descriptors i jerarquia comuna a totes les instàncies. De l'altra, definim Dimensions més petites amb Descriptors i jerarquies més específiques. Els usuaris podran utilitzar la Dimensió que més els interessi a cada moment per consultar el Fet.

El mateix pot passar amb els Fets. Alguns esdeveniments tindran més atributs o, fins i tot, més Dimensions que d'altres. De la mateixa manera que en el cas de les Dimensions, el que hem de fer és especialitzar el Fet.

Si només canvia el nombre d'atributs, podem mantenir una sola Estrella amb més d'un Fet. Per contra, si segons el tipus de Fet tenim més o menys Dimensions, és millor definir una Estrella diferent.

Figura 32



Exemple d'especialització d'un Fet

A la figura 32, podem veure dues Estrelles en les quals els Fets respectius estan relacionats per una especialització. Mentre que en una venda a crèdit podem identificar el client, quan el pagament és en efectiu aquesta identificació no és possible. Per tant, la Dimensió `Client` no està present a l'Estrella per estudiar les vendes en general.

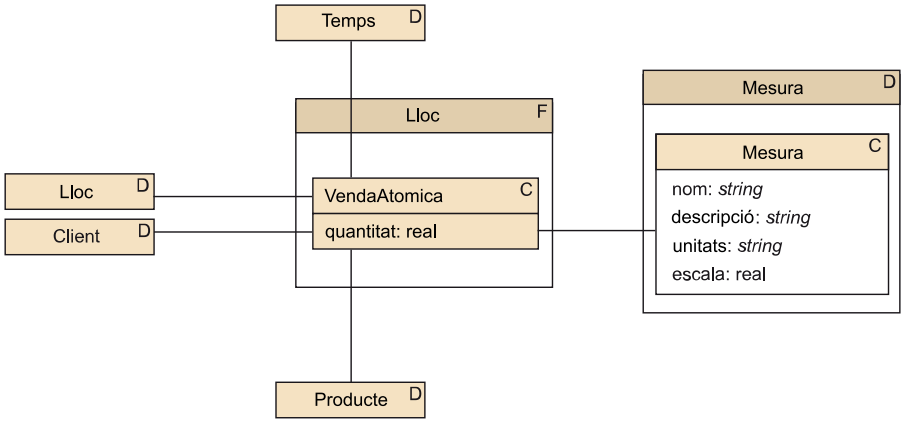
Quan les instàncies són heterogènies, podem especialitzar tant els Fets com les Dimensions per evitar l'aparició de Descriptors i Mesures invàlides, o facilitar la definició de jerarquies d'agregació o Dimensions específiques.

3.2.5. Fets amb una sola mesura

Un altre cas d'heterogeneïtat de les instàncies es produeix quan a cada esdeveniment prenem diferents mesures. Per especialitzar, cal diferenciar diferents tipus de Fets i que les Mesures siguin específiques de cada tipus concret. No obstant això, què podem fer quan les mesures que prenem depenen de l'esdeveniment que es produeix i no podem definir tipus d'esdeveniments?

El millor en aquests casos és definir una Dimensió `Mesura`. El Fet tindrà una sola Mesura, que podem denominar `quantitat`. Segons a quina instància de `Mesura` associem una instància del Fet, `quantitat` tindrà un significat o un altre. A més d'ajudar a solucionar problemes de diferències en les Mesures de cada esdeveniment, aquest tipus de disseny significa que les Mesures ens interessen de manera individual i no totes alhora (amb un disseny com aquest, no hauria de ser freqüent que consultéssim tots els mesuraments que vam fer quan es duia a terme un esdeveniment).

Figura 33



Exemple de Fet amb una Mesura

A la figura 33, podeu veure com hauríem de modificar la nostra Estrella de vendes perquè contingués una sola Mesura. Aprofitant que generem una nova Dimensió, podem afegir Descriptors de les mesures, com per exemple les unitats, l'escala, etc.

Com a opció de disseny, podem substituir totes les Mesures d'un Fet per una de sola, afegint una nova Dimensió `Mesura`.

4. Disseny lògic

Una vegada hem vist com es fa un disseny conceptual multidimensional, ara veurem com es passa al model lògic. Al mercat hi ha diferents tipus d'implementacions multidimensionals: ROLAP, MOLAP o, fins i tot, O^3LAP. A causa de la implantació actual dels sistemes relacionals, que cobreixen el mercat, l'opció més comuna és la implementació ROLAP.

ROLAP, MOLAP, O3LAP

Els tres conceptes provenen de l'anglès. *ROLAP* és *relational OLAP*, *MOLAP* és *multidimensional OLAP* i *O3LAP*, *object-oriented OLAP*.

Una eina ROLAP no és més que una capa de programari que rep consultes multidimensionals, les tradueix a SQL i les executa sobre un SGBD re- lacional. Ara veurem com s'implementa un esquema multidimensional sobre les taules d'un SGBD d'aquest tipus.

4.1. L'Estrella (el cas bàsic)

Igual que fèiem en el disseny conceptual, per passar al model lògic ens fixem també en una Estrella cada vegada. A més, en aquest apartat pensem que el Fet conté una única Cel·la. Per implementar una Estrella, necessitem una taula per al Fet (en el qual cada fila representa una cel·la de l'espai multidimensional) i una taula més per a cadascuna de les Dimensions. Les jerarquies d'agregació queden implícites en els valors dels atributs de les taules de Dimensió. No els explicitem amb taules diferents.

Exemple de taula de Dimensió amb jerarquia d'agregació implícita

La Dimensió Temps podria estar implementada amb la relació següent:

Temps(RowID,diaMesAny,mesAny,setmanaAny,any,festiu,laborable,traspas…)

Observeu que aquesta relació no està normalitzada. L'atribut `diaMesAny` determina tant `mesAny` com `setmanaAny`. A més, qualsevol d'aquests dos atributs determina `any`. D'aquesta manera, sabem a quin mes, setmana i any pertany cadascun dels dies sense necessitat de definir la jerarquia d'agregació. Les agrupacions possibles de les files queden determinades pels valors dels mateixos atributs, en lloc de per una jerarquia explícita. Totes les files amb el mateix valor en l'atribut `any` s'agrupen per donar lloc a una instància a Nivell `Any`.

Taula de Dimensió

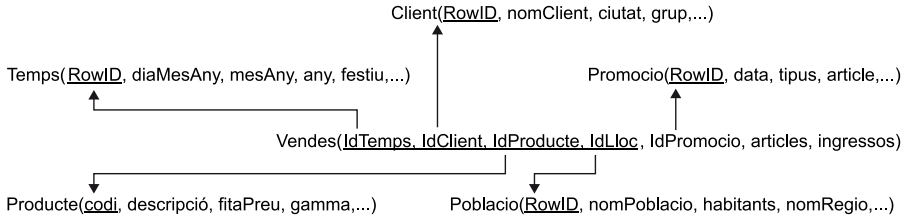
El terme anglès per a aquesta estructura de taules i claus foranes és *star join*.

La taula de Fet estarà relacionada per claus foranes amb les taules de Dimensió.

Cada clau forana apunta de la taula del Fet cap a una de les taules de les Dimensions. Juntament amb les claus foranes, la taula del Fet conté les Mesures, mentre que les taules de Dimensió contenen els Descriptors. Si el Fet no conté Mesures, la taula del Fet només contindrà les claus foranes cap a les taules de Dimensió.

Com a clau primària de la taula del Fet, tindrem els atributs corresponents a una de les Bases de la Cel·la atòmica. La resta de les Bases de la Cel·la donaran lloc a claus alternatives. En qualsevol cas, tant la clau primària com les alternatives seran subconjunts del conjunt de claus foranes que apunten cap a les taules de Dimensió.

Figura 34



Exemple d'esquema relacional amb forma d'estrella

En el cas de les vendes i les promocions, obtindríem l'esquema relacional de la figura 34. Podem veure que hi ha sis taules: una per a cada Dimensió (`Temps`, `Producte`, `Poblacio`, `Promocio` i `Client`) i una més per al Fet (`Vendes`). Aquestes taules estan relacionades per cinc claus foranes, cadascuna de les quals va de la taula del Fet a la clau primària (el codi de producte per a la taula `Producte` o el *RowID* per a la resta de les taules) d'una de les taules de les Dimensions. Quatre d'aquestes claus foranes formen la clau primària de la taula del Fet. Si considerem que {`Promocio`, `Client`, `Dia`, `Poblacio`} és una Base, llavors {`IDPromocio`, `IDClient`, `IDTemps` i `IDLloc`} seria clau alternativa de la taula Vendes. Els atributs `articles` i `ingressos`, les Mesures (que mai no són clau forana), no formen part de la clau primària de `Vendes`.

Nota

Recordeu que marquem la clau primària d'una relació subratllant els atributs que la formen.

Cada Estrella dona lloc a una taula per al Fet i a una més per a cadascuna de les Dimensions. Aquestes taules estan vinculades per claus foranes que van de la taula del Fet a cadascuna de les taules de Dimensió. La clau primària de la taula del Fet és la concatenació de les claus foranes corresponents a una Base del Fet.

Observeu la mida de cadascuna de les taules. La taula del Fet serà d'ordres de magnitud major que qualsevol de les taules de Dimensió. Ocuparà més d'un 95 % de l'espai utilitzat per l'Estrella. A primera vista, pot semblar estrany, però una manera de reduir la mida de la taula del Fet consisteix a definir substituts de la clau primària a les taules de Dimensió. Com que els substituts ocuparan menys espai que els atributs identificadors de la mateixa taula (per exemple, un DNI ocupa vuit caràcters, mentre que un *RowID* només quatre *bytes*), utilitzant-los reduïm la mida de les columnes que formen la clau primària de les taules de Dimensió. Per tant, també reduïm la mida de la clau primària de la taula del Fet, perquè sabem que sempre està formada per claus foranes que apunten a les claus primàries de les taules de Dimensió. Un petit guany a cada fila de la taula del Fet, que sol tenir milions de files, pot representar un gran estalvi d'espai.

Clau primària d'una tabla

Recordeu que, com a clau primària d'una taula, no només podem tenir atributs, sinó també substituts (*surrogates*), coneguts també com a *RowID*. Podeu veure el mòdul «Reconsideració dels models conceptual i lògic» de l'assignatura *Sistemes de gestió de bases de dades*.

L'efecte col·lateral de la utilització de substituts de la clau primària és que prevenim possibles problemes davant de canvis en els atributs identificadors en els sistemes operacionals. Recordeu que volem analitzar llargs períodes de temps. Per tant, és altament probable que durant aquest període es produei-

xin recodificacions dels identificadors de les promocions, de les poblacions o dels clients, amb la qual cosa també hauríem de recodificar totes les dades que tinguem a la nostra Estrella. Si utilitzem substituts per a la clau primària, ja no tenim aquest problema, perquè l'identificador és absolutament independent de l'origen de les dades i qualsevol canvi que es produeixi no l'afectarà.

Definim substituts de la clau primària a les taules de Dimensió per reduir la mida de la taula del Fet. A més, també serveix per evitar problemes si es modifiquen els identificadors en els sistemes operacionals.

A més de la mida, també hi ha diferències en les operacions que fem a cada tipus de taula. Les taules de Dimensió es creen amb les dades a dins i molt poques vegades canvien. Només de tant en tant s'afegeix una nova fila o es canvia el valor d'una que ja hi havia (mai no s'esborren files). A la taula de Fets, inserim files massivament de manera regular i només pateix modificacions si hem comès un error durant la inserció (tampoc no s'esborra mai res).

4.2. El floc de neu

Ja hem vist com es pot tractar el cas en el qual només tenim una cel·la. No obstant això, què cal fer quan en tenim més d'una dins del mateix Fet? De la mateixa manera que abans, definim una taula de Fet per a cada Cel·la, amb les corresponents claus foranes cap a les taules de Dimensió. Respecte a les taules de Dimensió, ara sí que cal explicitar els Nivells (normalitzar-les), encara que només parcialment.

Adoneu-vos que, si normalitzem totalment una taula de Dimensió, obtindrem una taula diferent per a cadascun dels Nivells. Fer-ho per a totes les Dimensions genera un esquema amb forma de floc de neu. El guany d'espai respecte de l'espai total de l'Estrella no resultaria significatiu, perquè les taules de Dimensió són insignificants respecte del volum de dades que conté la taula del Fet. Per contra, empitjoraria el rendiment de les consultes, perquè cada vegada que volguéssim operar amb una Dimensió hauríem de fer la combinació (*join*) de les taules de Dimensió corresponents als Nivells.

Contingut complementari

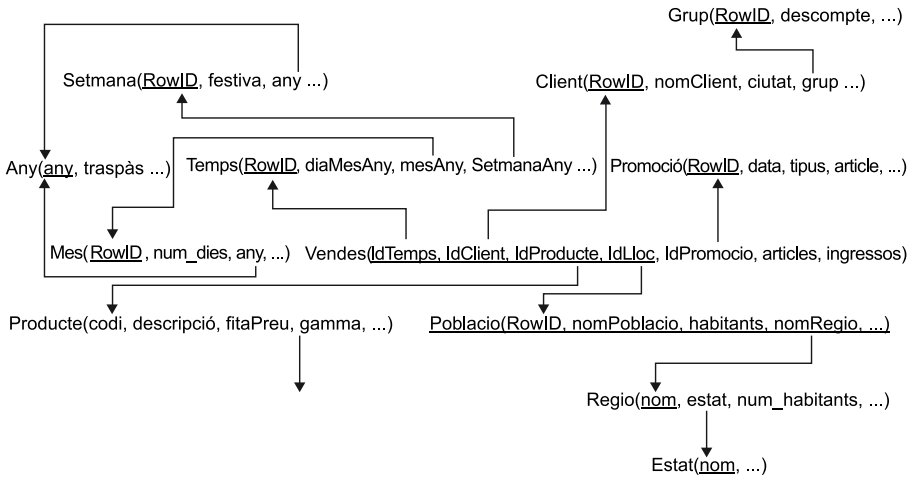
Estrella normalitzada

Una estrella completament normalitzada es coneix amb el nom de *floc de neu* (*snowflake*), per la seva forma d'estrella amb ramificacions, que recorda l'estructura d'un floc de neu.

La normalització és una eina del món transaccional que evita les redundàncies i facilita la concurrència en presència d'actualitzacions, però augmenta el nombre de combinacions necessàries per resoldre algunes consultes. Hem de pensar que l'objectiu d'un esquema multidimensional és respondre consultes de manera eficient (com ja hem dit, no hi ha actualitzacions), i resoldre combinacions no és gens ràpid. Amb les Dimensions desnormalitzades, podem millorar el rendiment d'algunes consultes fins a un 30%.

En la figura següent, podeu veure com quedaria l'esquema un cop normalitzat. Podeu apreciar que, a més dels problemes ja esmentats, ara l'esquema és més difícil d'entendre (sobretot per a usuaris no experts). Desgraciadament, a més de ser més difícil d'entendre per als usuaris, també ho és per a l'optimitzador de consultes. Alguns reconeixen una Estrella i utilitzen tècniques específiques per a la resolució de consultes, però no n'hi ha cap que reconegui un floc de neu.

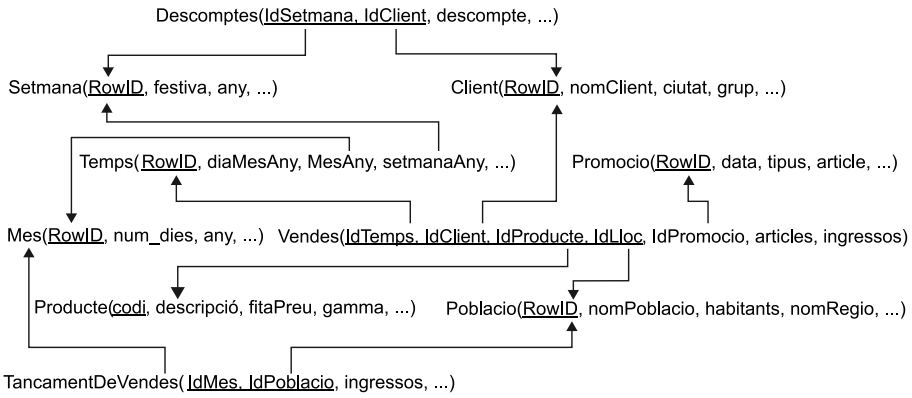
Figura 35



El floc de neu és un error de disseny que genera un guany inapreciable d'espai i una gran pèrdua de rendiment. Per millorar el rendiment de les consultes, hem d'evitar normalitzar els esquemes, excepte casos extrems en els quals la mida de la Dimensió sigui comparable amb la del Fet.

Només hem de definir taules de Dimensió per als Nivells que tenen alguna Cel·la associada. D'aquesta manera, les claus foranes de la taula del Fet apuntaran a la granularitat correcta. No és possible que les Mesures corresponguin a dades mensuals i la clau forana de l'associació amb la Dimensió `Temps` apunti a una taula que conté informació de dies.

Figura 36



Exemple de pas a relacional d'un Fet amb múltiples Cel·les

A la figura 36 podeu veure com quedaria un esquema relacional quan tenim diferents Cel·les en un mateix Fet. En aquest cas, tenim tres taules de Fet (`Descomptes`, `Vendes` i `TancamentDeVendes`). Cadascuna d'aquestes apunta a les seves taules de Dimensió. Us heu de fixar especialment en la normalització parcial que ha tingut la taula `Temps`. Ara hi ha dues taules més (`Setmana` i `Mes`) relacionades amb aquesta mitjançant claus foranes. Observeu que la informació de l'any ja no és a la taula `Temps`, sinó en les taules `Setmana` i `Mes` (repetida a totes dues). No s'ha acabat de normalitzar perquè no hi ha cap Cel·la en el Nivell `Any`.

Només normalitzarem una Dimensió quan el Fet contingui Cel·les diferents i sigui necessari fer-ho per relacionar la taula de Fet corresponent a cada Cel·la amb la granularitat adequada de la Dimensió.

4.3. Conformació: compartició de Dimensions

Conformar

Conformar significa fer quelcom conforme a una norma.

Generalment, cada Estrella té les seves Dimensions. No obstant això, és habitual que certes Dimensions s'utilitzin en diferents Estrelles (per exemple, la Dimensió `Temps` o `Clients`). Cal consensuar entre els analistes implicats quin ha de ser el contingut i format d'aquestes Dimensions compartides. Això es denomina conformar les Dimensions. D'aquesta manera, podrem utilitzar en diferents Estrelles taules de Dimensió que tinguin la mateixa forma, malgrat que puguin estar implementades sobre SGBD diferents o, fins i tot, sobre màquines diferents.

Conformació de Dimensions

Un bon indicador per identificar quan dues Dimensions que s'han de conformar és que es denominin igual. No obstant això, heu de vigilar amb això perquè, per exemple, la Dimensió temporal de vendes no es pot conformar amb la Dimensió temporal fiscal, la qual no ve marcada per les estacions o dies de festa, sinó pels tancaments de comptes i dates de declaració d'impostos.

Conformar les Dimensions resulta extremadament important en el disseny de les Estrelles, per permetre navegar de l'una a l'altra i canviar el tema objecte d'anàlisi. S'han de conformar les Dimensions que participin en Estrelles diferents per fer possible el *drill-across*.

Avantatges de tenir les Dimensions conformades:

Possibilita el *drill-across*.

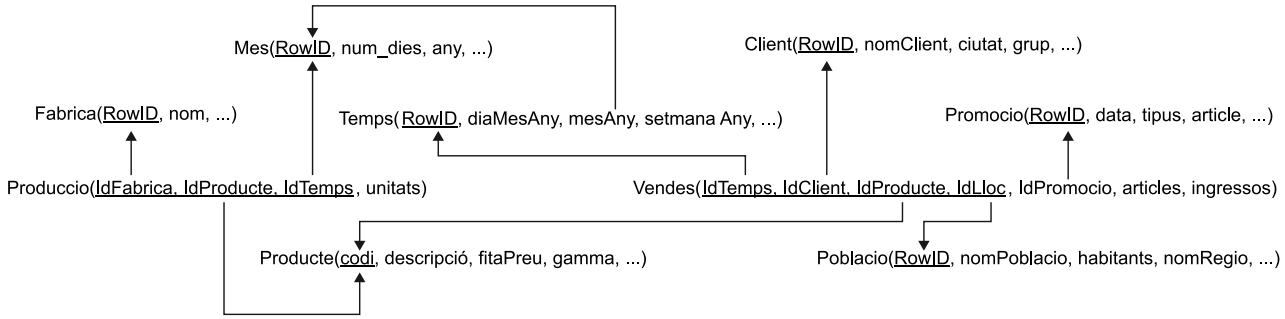
Estalvia treball de disseny i administració, perquè la mateixa taula s'utilitza més d'una vegada. Si implementéssim totes les Estrelles dins d'un mateix SGBD, fins i tot podríem tenir només una taula per a cada Dimensió, que seria compartida per tantes Estrelles com fos necessari.

Ajuda a consolidar la idea d'una factoria d'informació empresarial, en comptes de petits magatzems de dades departamentals aïllades. Tots els usuaris disposaran dels mateixos Descriptors i jerarquies d'agregació.

Una Dimensió conformada és la que s'utilitza en la implementació de més d'una Estrella. A l'hora de fer consultes, taules de Fet que comparteixen taules de Dimensió es poden substituir les unes per les altres (*drill-across*) sense modificar la resta de la consulta.

Compartir una Dimensió no significa que dues Estrelles hagin d'utilitzar exactament la mateixa taula de Dimensió. També es pot fer si una Estrella utilitza una taula que representa les dades de la Dimensió a un cert grau d'agregació i una altra Estrella utilitza una altra taula de Dimensió que representa un grau d'agregació superior. Si aquest és el cas, les dues taules formen part de la mateixa Dimensió.

Figura 37



Exemple de compartició de taules de Dimensió

A la figura 37 podem veure com `Vendes` i `Produccio` comparteixen Dimensions. En primer lloc, la taula de Dimensió `Producte` és apuntada per les dues taules de Fet. Per tant, cal haver conformat la Dimensió `Producte` de les dues Estrelles. A més, no obstant això, també tenen en comú la Dimensió `Temps`. En aquest cas, cada Estrella la utilitza amb una granularitat diferent. `Produccio` la utilitza amb granularidad `Mes`, mentre que `Vendes` l'utilitza amb granularitat `Dia`. Cal, doncs, haver conformat també aquesta Dimensió.

Conformar les Dimensions no implica que implementem totes les Estrelles en el mateix SGBD, i ni tan sols ho hem de fer a la mateixa màquina. El més important és que les taules que implementin la mateixa Dimensió en Estrelles diferents posseeixin la mateixa forma (els mateixos atributs i dominis semàntics) per permetre passar d'una taula de Fet a l'altra. Si aquestes taules es fusionen en una única taula o tenim una taula diferent a cada SGBD o a cada màquina, és un tema que no hem de decidir quan es fa el disseny lògic.

4.4. Dimensions degenerades, de rebuig i ombres

En alguns casos, podem estar davant de Dimensions que no tenen atributs ni jerarquies d'agregació, però que ens són útils simplement per identificar les instàncies del Fet. En aquests casos parlem de Dimensions degenerades, perquè no donen lloc a cap taula de Dimensió. Simplement tenim un atribut a la taula de Fet, que forma part de la clau, però que no és clau forana (no apunta a cap taula de Dimensió).

Exemple de Dimensió degenerada

L'exemple més habitual de Dimensió degenerada és el número que identifica la comanda. Malgrat no tenir cap atribut, ens resulta útil per identificar les vendes i agrupar totes les que s'han fet dins de la mateixa comanda. Aquesta Dimensió `Comanda` no té cap atribut, perquè els hi hem tret tots (data, client, etc.) per definir altres Dimensions d'anàlisi.

Les Dimensions degenerades solen tenir el seu origen en atributs identificadors de les bases de dades operacionals que coincideixen amb el grànul triat per l'Estrella en qüestió. És important conservar-los perquè permeten saber exactament d'on provenen les dades.

Una Dimensió degenerada és aquella que no dona lloc a una taula de Dimensió per falta d'atributs, però que sí s'utilitza per identificar les instàncies del Fet.

Un altre cas interessant es produeix a l'inrevés: quan hi ha atributs que descriuen el Fet, però que no corresponen a cap concepte en concret que ajudi a identificar-lo. Què hem de fer quan tenim un conjunt d'atributs (sovint booleans) que provenen dels sistemes operacionals, independents els uns dels altres? En aquest cas, cal crear una taula de Dimensió auxiliar que contingui tots aquests atributs, que no identificaran en cap cas les instàncies del Fet, però que utilitzarem per seleccionar-les i agregar-les.

Aquest tipus de taula de Dimensió es denomina **Dimensió de rebuig**. També relacionarem aquest tipus de taules de la taula del Fet amb una clau forana, però en aquest cas no formarà part de la clau primària.

Per contenir tots aquells atributs que provenen dels sistemes operacionals que no corresponguin a cap Dimensió en concret, crearem una taula de Dimensió especial, la clau de la qual no formarà part de la clau primària del Fet.

En les Dimensions, també hi ha atributs que molt probablement no ens serviran per seleccionar ni per definir jerarquies d'agregació (com per exemple la direcció, el número de telèfon o simplement un comentari a la Dimensió Client), però que potser voldrem afegir a la consulta per generar l'informe final just abans d'imprimir-lo. Per no afectar el temps de resposta, podem posar tots aquests atributs poc utilitzats en una altra taula, que denominarem **Dimensió ombra**.

Aquesta Dimensió ombra no estarà relacionada directament amb la taula de Fet, sinó que tindrem a la Dimensió de debò una clau forana que l'apuntarà. D'aquesta manera, només caldrà accedir a aquests atributs (que habitualment ocupen molt espai) per imprimir l'informe final, i no afectaran el temps de resposta durant la navegació per les dades.

Una Dimensió ombra és aquella taula que conté atributs descriptius poc utilitzats i que no relacionem directament amb la taula de Fet, sinó amb una altra taula de Dimensió.

4.5. Generalitzacions/especialitzacions

Durant el disseny conceptual, quan hi ha instàncies heterogènies, ho representem amb una especialització del Fet o de la Dimensió, segons sigui el cas. Recordeu les tres opcions que tenim per implementar aquest tipus de vincle entre classes:

1) Una taula per a la superclasse i una altra per a cada subclasse, totes amb la mateixa clau primària.

2) Una única taula que contingui tant els atributs de la superclasse com els de totes les subclasses.

3) Només una taula per a cadascuna de les subclasses, repetint els atributs de la superclasse a cada taula.

En el disseny de bases de dades operacionals, la millor opció és la primera (encara que en algun cas molt especial, podria interessar més alguna de les altres dues). No obstant això, en el cas del disseny multidimensional no és així.

Si l'especialització és d'una Dimensió (com per exemple, la de la figura 31), cal considerar si les taules estaran referenciades per alguna taula de Fet o no. De la mateixa manera que no normalitzem per evitar haver de fer combinacions, ara només separarem les subclasses i superclasse en taules diferents quan sigui estrictament necessari.

Si la taula corresponent a la superclasse no és apuntada per una clau forana des de cap Fet i les corresponents a les subclasses sí que ho són, el millor és la tercera opció, amb la qual ens estalviarem combinacions. De la mateixa manera, si les taules corresponents a les subclasses no són apuntades per cap taula de Fet, la millor opció serà la segona. Per exemple, en el cas de la figura 31, si les Dimensions `Neveres` i `Vídeos` no s'utilitzen en cap altra Estrella, seria millor tenir només la taula de Dimensió corresponent a `Producte`, que en aquesta taula també inclou els atributs específics de neveres i vídeos.

Si l'especialització és d'una Dimensió, la implementarem amb una taula de Dimensió per a cadascuna de les classes que hi ha associades a algun Fet.

Si el que ha especialitzat és un Fet (com teniu dibuixat a la figura 32), atesa la quantitat de files que té una taula de Fet, la segona opció queda descartada directament per la quantitat de valors nuls que genera (amb el consegüent malbaratament d'espai i temps de consulta).

En canvi, malgrat que pugui malbaratar una mica d'espai, la tercera opció serà la millor, si cada analista està interessat en una única subclasse i vol tractar junts tant els atributs propis de la subclasse com els heretats de la superclasse. Amb aquesta opció, obtindríem diferents taules de Fet (relativament petites) per a cada analista, amb el consegüent guany de rendiment. Fer la unió de totes aquestes per obtenir la superclasse sempre és possible. No obstant això, cal dir que si normalment no es vol accedir al mateix temps als atributs de la superclasse i de les subclasses, la millor opció per implementar una especialització d'un Fet és la primera (una taula de Fet per a cada subclasse i una altra per a la superclasse).

Si l'especialització és d'un Fet, hem d'observar si s'accedeix als atributs de superclasse i subclasses al mateix temps o no. En el cas que la majoria de les vegades s'hi accedeixi al mateix temps, és millor que només tinguem taules per a les subclasses. Quan és més habitual accedir-hi per separat, llavors convé tenir els atributs de la superclasse en una taula diferent.

4.6. Estructures temporals

Ja hem dit que la Dimensió temporal està pràcticament en totes les Estrelles. Els Fets estan associats a la Dimensió temporal precisament perquè registra l'evolució del negoci i els seus canvis. Cada vegada que es produeix un esdeveniment, afegim una nova instància del Fet. Aquestes instàncies corresponen a successos concrets (per exemple, vendes i moviments en el compte corrent) o a estats (per exemple, estocs i saldos).

Vist d'una altra manera, hi ha dues possibilitats de registrar l'evolució temporal: guardar les transaccions (diferencials) o guardar successions de fotografies. En el primer cas, l'esdeveniment que cal registrar prové com a conseqüència d'una certa acció en el món real de la qual guardem els seus valors associats. En el segon cas, cada cert temps registrem l'estat del nostre Fet. D'una banda, podem registrar cada venda concreta i, de l'altra, podem registrar en un moment donat quant s'ha venut.

Els dos mecanismes ens permeten mostrar l'evolució del negoci, malgrat que el que es basa en transaccions és molt més detallat. A partir d'aquest podem construir les fotografies, però no a l'inrevés. Tot i així, els dos mecanismes són necessaris. Les transaccions ens donen el màxim grau de detall, però les fotografies ens permeten saber ràpidament l'estat de l'empresa en un moment donat.

Hi ha dues maneres de gravar els Fets. Ho podem fer mitjançant transaccions entre estats o per mitjà de fotografies de l'estat en un determinat moment.

Els Fets sempre registren els canvis que es produeixen en el negoci. I les Dimensions mai no registren cap canvi? No es modifiquen mai? Una màxima budista diu que l'única constant és que tot canvia. Per tant, no ens hem de preguntar si les Dimensions canvien, sinó amb quina freqüència ho fan.

Efectivament, les Dimensions canvien molt poc, però també tenen alguns canvis (obrirem noves botigues, canviarem el descompte que fem de manera habitual a un bon client, deixarem de vendre algun producte, etc.). Si els canvis són realment poc freqüents, una possible solució consisteix a definir una Estrella diferent cada vegada que es produeixi un canvi en una de les seves Dimensions. Una altra solució quan la Dimensió que canvia no és gaire gran consisteix a definir diferents versions de la taula de Dimensió i vincular-les totes al mateix cub. En aquest cas, l'usuari (o la interfície que utilitza el cub) ha de saber quina taula ha d'utilitzar per resoldre la consulta a cada moment. La millor solució i la més general, no obstant això, consisteix a gravar els canvis dins de la mateixa taula de Dimensió afectada.

En els sistemes operacionals, registrem un canvi simplement modificant el registre que toca. En un sistema d'anàlisi, això normalment no es pot fer d'aquesta manera. Imaginem-nos que canvia el preu d'un producte. Si senzillament canviem el valor de l'atribut `preu` a la taula de Dimensió, semblarà que totes les vendes s'han fet amb aquest preu. No podrem veure la modificació que es produeix en el volum de vendes segons quin sigui el preu del producte.

Quan veiem que una Dimensió canvia, tenim tres opcions:

1) Sobreescriure el valor antic amb el nou (com es fa en els sistemes operacionals). Observeu que si fem això, renunciem a estudiar com afecta aquest canvi al Fet. D'altra banda, simplifica molt el tractament i no utilitza més espai de l'estrictament necessari. No obstant això, aquesta opció només és aconsellable per als casos en els quals descobrim que havíem comès un error en la introducció de dades i el volem esmenar sense deixar-ne constància. També ho és quan l'atribut en concret no té cap incidència en l'anàlisi (per exemple, el

nom d'una companyia). Si té incidència en l'anàlisi, perquè defineix una característica utilitzada en la selecció o agrupació d'instàncies (com per exemple els habitants d'una població), aquesta opció no resulta gens adequada (podríem seleccionar Fets en una població amb un cert nombre d'habitants en un moment en el qual encara no els tenia).

2) Crear una nova fila a la taula de Dimensió (amb la seva *RowID*) que serà referenciada des d'ara per totes les instàncies de Fet que afegim. Aquesta és la solució més habitual. Observeu que implica la utilització d'un substitut de la clau primària a la taula de Dimensió. Com podríem identificar la segona fila dins de la taula? Si utilitzéssim el DNI com a clau primària, no podríem registrar així els canvis en les dades d'una persona, perquè les dues files haurien de tenir el mateix DNI.

Una altra raó per utilitzar substituts de la clau primària a les taules de Dimensió consisteix a registrar els canvis sense tenir problemes d'identificació.

Observeu que amb aquesta solució, el que fem realment és crear una nova instància de la Dimensió. En certa manera, diem que hi ha dues persones diferents, una abans del canvi i una altra després. Per fer consultes, no cal ser conscients del canvi. La mateixa condició ja indicarà quina de les dues files es tria i, per tant, en quins Fets estem interessats.

Un problema d'aquesta opció és que no serà evident quines dues files fan referència a la mateixa instància en moments diferents. Això es pot solucionar mantenint (a més del *RowID*) un atribut identificador (com per exemple el DNI) i afegir un atribut més que sigui el número de versió de la instància. Tindrem tantes versions d'una instància com canvis hi hagi en els seus atributs.

3) Tenir diferents atributs a la taula de Dimensió per registrar el valor antic i el nou. És a dir, per a cada atribut que pugui tenir un canvi, ara utilitzem dos atributs. Aquesta solució, com ja us podeu imaginar, és molt limitada i útil solo en casos molt concrets. En primer lloc, cal que el canvi sigui general a totes les files de la taula. No obstant això, a més, només funciona si hi ha un únic canvi. Si n'hi hagués dos, necessitaríem tres atributs; si n'hi hagués tres, quatre atributs, etc. Una implementació com aquesta pot resultar útil, per exemple, per canviar els noms de les poblacions de l'espanyol al català (o de l'espanyol al basc). Les poblacions són exactament les mateixes (no generem nous *RowID*). Simplement, volem utilitzar la nova denominació i no volem perdre la denominació espanyola. Amb aquesta opció, en alguns casos, a més de l'atribut amb el valor antic també cal un altre atribut amb la data del canvi.

Bàsicament, hi ha tres possibilitat per registrar els canvis a les taules de Dimensió:

- Modificar el valor directament.
- Crear una nova fila.
- Crear una nova columna.

Recordeu les minidimensions de les quals parlàvem a l'apartat de disseny conceptual. Observeu que definíem perfils d'usuari en lloc de usuaris pròpiament. En aquest tipus de Dimensió, no reflectirem els canvis de cap d'aquestes tres maneres. Si canvia un dels atributs de l'usuari, simplement associarem els seus esdeveniments amb un perfil diferent. No obstant això, la taula de Dimensió no tindrà cap modificació tret que aquest canvi generi un nou perfil d'usuari que no hi havia abans. Si és aquest el cas, simplement l'afegirem ara a la taula.

Aquestes minidimensions també ajuden a gestionar els canvis en les Dimensions. Normalment, tenim atributs que canvien amb més freqüència que d'altres. L'edat o els ingressos anuals tenen una freqüència de canvi molt més alta que la població de residència, el nom o el sexe. Per no haver d'afegir una nova fila a la taula Client cada vegada que canviïn aquests atributs, els podem posar dins d'una minidimensió demogràfica, en la qual ja hem dit que no cal afegir noves files cada vegada que es produeix un canvi, sinó simplement relacionar cada instància del Fet amb el perfil corresponent.

Els canvis en les minidimensions són més fàcils de gestionar que en les Dimensions normals, perquè no cal modificar-les. En alguns casos, crear-les pot ser una bona opció per registrar els canvis.

Generalment, podeu distribuir els atributs de qualsevol Dimensió segons la freqüència de canvi, per facilitar la seva gestió. D'aquesta manera, tindríeu una taula de Dimensió que no canvia gairebé mai i una altra que canvia cada cert temps.

Suposant que tenim definida la taula Client amb aquesta extensió: **Client(RowID, DNI, nom, edat, edatAnterior)** i que abans de l'actualització conté una fila amb els valors:

<111, 12345678, Jordi, 29, *null*>

Si es modifica el valor de la columna edat directament, informant-lo amb el valor 30 quedaria:

<111, 12345678, Jordi, 30, *null*>

Si, en canvi, decidíssim crear una nova columna per conservar l'edat anterior, la taula client hauria de tenir aquesta estructura: **Client(RowID, DNI, nom, edat, edatAnterior)**. En aquest cas, si abans de l'actualització tinguéssim la fila:

<111, 12345678, Jordi, 29, *null*>

Després tindríem:

<111, 12345678, Jordi, 30, 29>

Opcionalment, també podríem afegir la data d'actualització. L'extensió de la taula Client seria: **Client(RowID, DNI, nom, edat, edatAnterior, dataActEdat)**. Si aquesta taula contingués la fila:

<111, 12345678, Jordi, 29, *null*, *null*> Després de l'actualització, tindríem:

<111, 12345678, Jordi, 30, 29, 30/09/2019>

5. Consultes amb SQL'99

Ara que ja hem vist com s'implementen les estrelles en un SGBD relacional, vegem com les podem consultar amb SQL estàndard. A més de veure l'estructura bàsica d'una consulta, també observarem les clàusules especials que incorpora l'especificació de l'estàndard de 1999 (SQL'99).

5.1. Estructura bàsica de la consulta

El que tenim en un esquema amb forma d'estrella és una taula de Fet i una taula per a cadascuna de les Dimensions. Per fer una consulta, hem de posar totes aquestes taules a la clàusula FROM. A WHERE relacionarem cada taula de Dimensió amb la taula de Fet utilitzant la clau forana corresponent (mai no vincularem Dimensions entre si) i afegirem les condicions que volem sobre els atributs de les taules de Dimensió. A SELECT posarem les Mesures de la taula de Fet que vulguem visualitzar amb la corresponent operació d'agregació. Finalment, posarem la clàusula GROUP BY amb els identificadors dels Nivells en els quals volem veure les dades a cada Dimensió. Sempre afegirem els atributs que apareguin a GROUP BY a SELECT per distingir les files. Habitualment, també s'afegeix la clàusula ORDER BY amb tots els atributs de les taules de Dimensió que tenim a SELECT, per obtenir el resultat ordenat i facilitar-ne la visualització.

```
SELECT Nivell_1, ..., Nivell_n, Operacio(Mesura), ...
FROM Fet, Dimensio_1, ..., Dimensio_n
WHERE Fet=Dimensio_1 AND ... AND Fet=Dimensio_n
AND Descriptor^1=valor AND ... AND Descriptor_m=valor
GROUP BY Nivell_1, ..., Nivell_n
ORDER BY Nivell_1, ..., Nivell_n
```

Cadascuna de les operacions del model multidimensional que hem vist a l'apartat corresponent té una relació directa amb aquesta estructura de consulta.

- Selecció: per seleccionar uns punts o uns altres del nostre espai *n*-dimensional, el que hem de fer és afegir o treure condicions sobre els Descriptors a la clàusula WHERE.

- Projecció: traient-les de SELECT, deixem de veure les Mesures que no ens interessen.

- *Roll-up*: per augmentar o disminuir el nivell de detall amb el qual veiem les dades, només cal agrupar segons l'atribut que identifica el Nivell o un altre de la Dimensió corresponent. Cal posar els atributs corresponents a GROUP BY. En el cas de voler fer *roll-up* fins al Nivell `All`, el que fa falta és treure tots els atributs de la Dimensió.

- *Drill-across*: si volem canviar el tema d'anàlisi, només cal substituir la taula de Fet a la clàusula FROM (recordeu que per poder-ho fer cal que els dos Fets comparteixin Dimensions). Les Dimensions que no hi ha en el nou Fet desapareixeran.

- CanviBase: finalment, si el que volem és veure les mateixes dades ordenades de manera diferent, només cal modificar els atributs de SELECT i canviar l'ordre dels atributs a ORDER BY.

Exemple de seqüència d'operacions sobre una consulta SQL

Atesa la consulta:

```
SELECT d1.nom_article, d2.nom_fabrica, d3.mesAny,
     SUM(fet.unitats)
FROM Produccio fet, Producte d1, Fabrica d2, Temps d3
WHERE fet.IDProducte=d1.id
     AND fet.IDFabrica=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs', 'Gomes')
     AND d2.num_treballadors>100
     AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d1.nom_article, d2.nom_fabrica, d3.mesAny
ORDER BY d1.nom_article, d2.nom_fabrica, d3.mesAny;
```

$A := roll\text{-}up_{Fàbrica::All}$("Unitats produïdes per Producte, Fàbrica i Mes")

```
SELECT d1.nom_article, 'All', d3.mesAny, SUM(fet.unitats)
FROM Produccio fet, Producte d1, Fabrica d2, Temps d3
WHERE fet.IDProducte=d1.id
     AND fet.IDFabrica=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs', 'Gomes')
     AND d2.num_treballadors>100
     AND d3.mesAny IN ('Gener2002','Febrer2002')
     GROUP BY d1.nom_article, d3.mesAny
ORDER BY d1.nom_article, d3.mesAny;
```

$B := canviBase_{Producte \times Temps}(A)$

```
SELECT d1.nom_article, d3.mesAny, SUM(fet.unitats)
FROM Produccio fet, Producte d1, Fabrica d2, Temps d3
WHERE fet.IDProducte=d1.id
     AND fet.IDTemps=d3.id
     AND fetIDFabrica=d2.id
     AND d1.nom_article IN ('Bolígrafs', 'Gomes')
     AND d2.num_treballadors>100
     AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d1.nom_article, d3.mesAny
ORDER BY d1.nom_article, d3.mesAny;
```

$C := drill\text{-}across_{Vendes}(B)$

```
SELECT d1.nom_article, d3.mesAny, SUM(fet.articles),
     SUM(fet.ingressos)
FROM Vendes fet, Producte d1, Temps d3
WHERE fet.IDProducte=d1.id
     AND fet.IDTemps=d3.id
```

```
      AND d1.nom_article IN ('Bolígrafs', 'Gomes')
      AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d1.nom_article, d3.mesAny
ORDER BY d1.nom_article, d3.mesAny;
```

$$D := projecció_{articles}(C)$$

```
SELECT d1.nom_article, d3.mesAny, SUM(fet.articles)
FROM Vendes fet, Producte d1, Temps d3
WHERE fet.IDProducte=d1.id
      AND fet.IDTemps=d3.id
      AND d1.nom_article IN ('Bolígrafs', 'Gomes')
      AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d1.nom_article, d3.mesAny
ORDER BY d1.nom_article, d3.mesAny;
```

$$E := canviBase_{Articles \times Lloc \times Temps}(D)$$

```
SELECT d1.nom_article, 'All', d3.mesAny, SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id
      AND fet.IDLloc=d2.id
      AND fet.IDTemps=d3.id
      AND d1.nom_article IN ('Bolígrafs', 'Gomes')
      AND d3.mesAny IN ('Gener2002','Febrer2002)
GROUP BY d1.nom_article, d3.mesAny
ORDER BY d1.nom_article, d3.mesAny;
```

$$F := drill\text{-}down_{Lloc::Regió}(E)$$

```
SELECT d1.nom_article, d2.regio, d3.mesAny, SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id
      AND fet.IDLloc=d2.id
      AND fet.IDTemps=d3.id
      AND d1.nom_article IN ('Bolígrafs', 'Gomes')
      AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d1.nom_article, d2.regio, d3.mesAny
ORDER BY d1.nom_article, d2.regio, d3.mesAny;
```

$$R := selecció_{Regió.nom="Catalunya"}(F)$$

```
SELECT d1.nom_article, d2.regio, d3.mesAny, SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id
      AND fet.IDLloc=d2.id
      AND fet.IDTemps=d3.id
      AND d1.nom_article IN ('Bolígrafs', 'Gomes')
      AND d2.regio='Catalunya'
      AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d1.nom_article, d2.regio, d3.mesAny
ORDER BY d1.nom_article, d2.regio, d3.mesAny;
```

5.2. GROUPING SETS

El més habitual és que els usuaris no només vulguin veure unes determinades dades, sinó també els totals. És a dir, el resultat d'una consulta seria una taula com aquesta:

Taula 5

Vendes	**Catalunya**		
	Gener 2002	**Febrer 2002**	**Total**
Bolígrafs	275.827 (a)	290.918 (a)	566.745 (c)

Vendes	Catalunya		
	Gener 2002	Febrer 2002	Total
Gomes	784.172 (a)	918.012 (a)	1.702.184 (c)
Total	105.999 (b)	1.208.930 (b)	2.268.929 (d)

Aquesta taula no és pròpiament un cub, perquè barreja cel·les de quatre granularitats diferents (les a, les b, les c i la d). Així i tot, la podem entendre com la unió de quatre cubs. De la mateixa manera que podem definir una funció per trossos, també podem definir la taula per trossos unint cubs.

Vendes (a) ⊕*Roll-upTemps::All*(*Vendes*) (b) ⊕*Roll-upArticles::All*(*Vendes*) (c) ⊕*Roll-upArticles::All,Temps::All*(*Vendes*) (d)

Observeu que la consulta SQL que hem vist abans només ens mostra les quatre cel·les (a). Per aconseguir les altres cinc cel·les seria necessari quatre consultes, com es fa a la consulta següent:

(a)

```
SELECT d1.nom_article, d2.regio, d3.mesAny, SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs','Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d1.nom_article, d2.regio, d3.mesAny
UNION
```

(b)

```
SELECT d1.nom_article, d2.regio, 'Total', SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id
     AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs','Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d1.nom_article, d2.regio
UNION
```

(c)

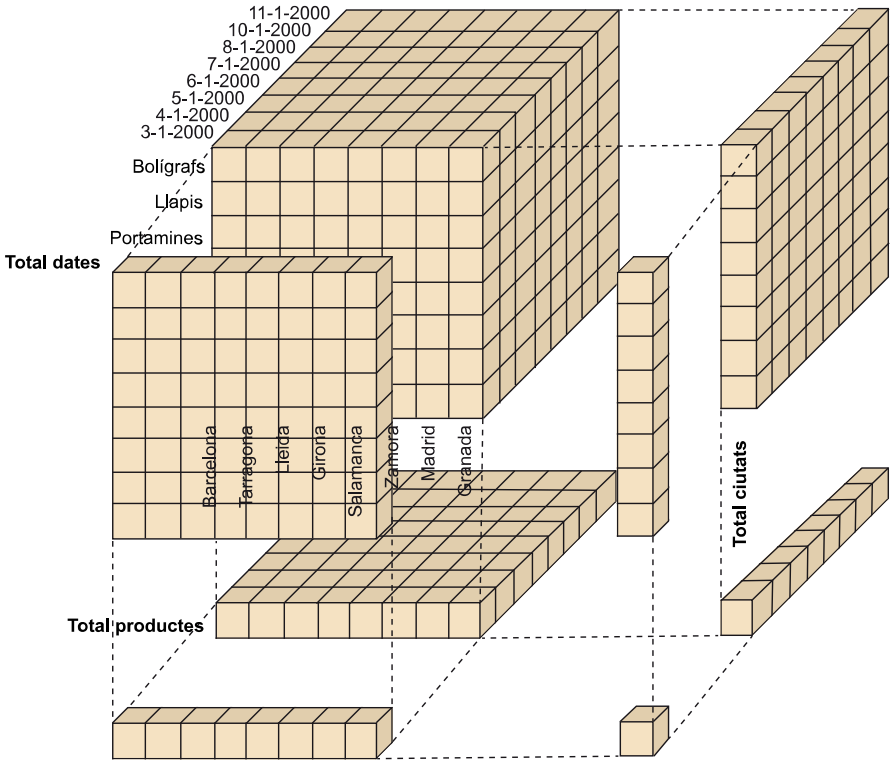
```
SELECT 'Total', d2.regio, d3.mesAny, SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id
     AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs','Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d2.regio, d3.mesAny
UNION
```

(d)

```
SELECT 'Total', d2.regio, 'Total', SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id
     AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs','Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002')
GROUP BY d2.regio
ORDER BY d1.nom_article, d2.regio, d3.mesAny;
```

Si en lloc de calcular el total per a dues Dimensions el volguéssim fer per a tres, necessitaríem set unions. La figura següent us mostra esquematitzats quins són els vuit cubs que caldria unir (vendes per dia, producte i ciutat; per dia i producte; per dia i ciutat; per producte i ciutat; per dia; per producte; per ciutat, i el total de vendes).

Figura 38



El nombre d'unions que cal fer per calcular els totals creix de manera exponencial respecte al nombre de Dimensions que tinguem. Per sort, l'estàndard SQL'99 ja ens ofereix una altra sintaxi per abreujar totes aquestes unions. Com podeu veure en aquesta consulta, només cal posar a GROUP BY les paraules clau 'GROUPING SETS' i, entre parèntesi, la llista d'agrupacions que es vol fer.

```
SELECT d1.nom_article, d2.regio, d3.mesAny, SUM(fet.article)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProdcute=d1.id
     AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs', 'Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002)
```

```
GROUP BY GROUPING SETS ((d1.nom_article, d2.regio, d3.mesAny),
                        (d1.nom_article, d2.regio),
                        (d2.regio, d3.mesAny),
                        (d2.regio))
ORDER BY d1.nom_article, d2.regio, d3.mesAny;
```

Per calcular diferents agrupacions de la mateixa consulta sense haver d'explicitar les unions, l'estàndard SQL'99 ens ofereix les paraules reservades GROUPING SETS dins de la clàusula GROUP BY.

El resultat de la consulta anterior seria aquest:

Nom_article	Regió	MesAny	Articles
Bolígrafs (a)	Catalunya (a)	Gener02 (a)	275827 (a)
Bolígrafs (a)	Catalunya (a)	Febrer02 (a)	290918 (a)
Bolígrafs (c)	Catalunya (c)	NULL (c)	566745 (c)
Gomes (a)	Catalunya (a)	Gener02 (a)	784172 (a)
Gomes (a)	Catalunya (a)	Febrer02 (a)	918012 (a)
Gomes (c)	Catalunya (c)	NULL (c)	1702184 (c)
NULL (b)	Catalunya (b)	Gener02 (b)	1059999 (b)
NULL (b)	Catalunya (b)	Febrer02 (b)	1208930 (b)
NULL (d)	Catalunya (d)	NULL (d)	2268929 (d)

A part del problema evident de la presentació, que ha d'arreglar la mateixa interfície gràfica, observeu els valors nuls. Què signifiquen, 'desconegut' o 'inaplicable'? Doncs cap de les dues opcions. Aquest és un tercer significat dels valors nuls que s'ha definit en el SQL'99. En aquesta taula, volen dir 'total'.

El valor nul no només significa 'desconegut' o 'inaplicable'. En el context dels GROUPING SETS, també pot voler dir 'total'.

Això planteja un nou problema. Com sabem si un valor nul significa que no coneixem el valor de l'atribut o que la fila és el resultat d'aplicar una operació d'agregació? Per contestar a aquesta pregunta, l'estàndard defineix la funció GROUPING.

La funció GROUPING té com a paràmetre un atribut. Retorna «1» si aquest atribut pren el valor 'total'. En qualsevol altre cas, retorna «0». Amb aquesta funció, podem reescriure la consulta anterior de manera que aparegui la paraula total en lloc dels valors nuls sempre que convingui (atès que d2.region apareix en les quatre agrupacions, sabem que no pot prendre el valor 'total').

```
SELECT
     CASE WHEN GROUPING(d1.nom_article)=1
          THEN 'TotalDeBolígrafsIGomes'
          ELSE d1.nom_article, d2.regio,
     CASE WHEN GROUPING(d3.mesAny)=1
          THEN 'TotalDeGenerIFebrer'
          ELSE d3.mesAny,
     SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProdcute=d1.id
     AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs', 'Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002)
GROUP BY GROUPING SETS ((d1.nom_article, d2.regio, d3.mesAny),
                        (d1.nom_article, d2.regio),
                        (d2.regio, d3.mesAny),
                        (d2.regio))
ORDER BY d1.nom_article, d2.regio, d3.mesAny;
```

Ara, el resultat de la consulta seria aquest:

Nom_article	Regió	mesAny	Articles
Bolígrafs (a)	Catalunya (a)	Gener02 (a)	275827 (a)
Bolígrafs (a)	Catalunya (a)	Febrer02 (a)	290918 (a)
Bolígrafs (c)	Catalunya (c)	TotalDeGenerIFebrer (c)	566745 (c)
Gomes (a)	Catalunya (a)	Gener02 (a)	784172 (a)
Gomes (a)	Catalunya (a)	Febrer02 (a)	918012 (a)
Gomes (c)	Catalunya (c)	TotalDeGenerIFebrer (c)	1702184 (c)
TotalDeBolígrafsIGomes (b)	Catalunya (b)	Gener02 (b)	1059999 (b)
TotalDeBolígrafsIGomes (b)	Catalunya (b)	Febrer02 (b)	1208930 (b)
TotalDeBolígrafsIGomes (d)	Catalunya (d)	TotalDeGenerIFebrer (d)	2268929 (d)

Podeu distingir el significat d'un valor nul utilitzant la funció GROUPING.

5.2.1. ROLLUP

Com ja hem dit, el nombre de totals creix de manera exponencial respecte al nombre de Dimensions. Per tant, encara que no hàgim d'escriure tota la consulta, escriure només totes les combinacions d'atributs a GROUPING SETS ja pot arribar a resultar massa complex. Per facilitar encara més aquest tipus de consulta, l'estàndard també defineix la paraula reservada ROLLUP. Amb un determinat conjunt d'atributs, calcula totes les agrupacions que resulten d'anar

L'ordre dels atributs

L'ordre dels atributs d'agrupació dins de ROLLUP sí que afecta el resultat de la consulta. L'ordre dels atributs dins de ORDER BY no afecta el seu resultat.

ignorant els atributs un per un, de dreta a esquerra. Vegem com s'incorporaria a la consulta anterior (presteu atenció a l'ordre dels atributs a GROUP BY i a ORDER BY).

```
SELECT d1.nom_article, d2.regio, d3.mesAny,
     SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id
     AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs', 'Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002)
GROUP BY ROLLUP (d2.regio, d1.nom_article, d3.mesAny);
ORDER BY d2.regio, d3.mesAny, d1.nom_article;
```

El resultat d'aquesta consulta seria el següent:

Nom_article	Regió	mesAny	articles
Bolígrafs (a)	Catalunya (a)	Gener02 (a)	275.827 (a)
Gomes (a)	Catalunya (a)	Gener02 (a)	784.172 (a)
Bolígrafs (a)	Catalunya (a)	Febrer02 (a)	290.918 (a)
Gomes (a)	Catalunya (a)	Febrer02 (a)	918.012 (a)
Bolígrafs (c)	Catalunya (c)	NULL (c)	566.745 (c)
Gomas (c)	Catalunya (c)	NULL (c)	1.702.184 (c)
NULL (d)	Catalunya (d)	NULL (d)	2.268.929 (d)
NULL (d)	NULL (d)	NULL (d)	2.268.929 (d)

Les primeres quatre files corresponen a «GROUP BY d2.regio, d1.nom_article, d3.mesAny»; les dues següents a «GROUP BY d2.regio, d1.nom_article»; la següent a «GROUP BY d2.regio»; i l'última a «GROUP BY ()». Com que només disposem d'un valor per a `d2.region`, les dues últimes files tenen el mateix valor. Podem evitar que surti l'última fixant aquest atribut i indicant que s'utilitzi a totes les agrupacions.

Nueva sintaxis

```
SELECT COUNT(*)
FROM taula;
```

Amb SQL'99, ara també es pot escriure:

```
SELECT COUNT(*)
FROM taula
GROUP BY ();
```

```
SELECT d1.nom_article, d2.regio, d3.mesAny, SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs', 'Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002)
GROUP BY d2.regio, ROLL-UP (d1.nom_article, d3.mesAny);
ORDER BY d2.regio, d3.mesAny, d1.nom_article;
```

El resultat d'aquesta consulta és el mateix d'abans, menys l'última fila. Amb això, podem reescriure la consulta original de quatre agrupacions de la manera següent:

```
SELECT d1.nom_article, d2.regio, d3.mesAny, SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id
```

```
     AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs', 'Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002)
GROUP BY GROUPING SETS
                 ((d2.regio, ROLL-UP (d1.nom_article, d3.mesAny)),
               (d2.regio, d3.mesAny))
ORDER BY d1.nom_article, d2.regio, d3.mesAny;
```

```
GROUP BY ROLL-UP (a1,..,an)
```

equival a:

```
GROUP BY GROUPING SETS ((a1,..,an),
                        (a1,..,an-1),
                        ...
                        (a1),
                        ())
```

5.2.2. CUBE

Amb ROLLUP ja no fa falta que escrivim totes les combinacions d'atributs que ens interessin, però encara n'hem d'escriure algunes. Les podem aconseguir totes directament si utilitzem la paraula reservada *CUBE* en comptes de *ROLLUP*. Amb la consulta següent, obtenim les vuit cel·les que volíem originalment:

```
SELECT d1.nom_article, d2.regio, d3.mesAny, SUM(fet.articles)
FROM Vendes fet, Producte d1, Lloc d2, Temps d3
WHERE fet.IDProducte=d1.id
     AND fet.IDLloc=d2.id
     AND fet.IDTemps=d3.id
     AND d1.nom_article IN ('Bolígrafs', 'Gomes')
     AND d2.regio='Catalunya'
     AND d3.mesAny IN ('Gener2002','Febrer2002)
GROUP BY d2.regio, CUBE (d1.nom_article, d3.mesAny);
ORDER BY d1.nom_article, d2.regio, d3.mesAny;
```

```
GROUP BY CUBE (a,b,c)
```

equival a:

```
GROUP BY GROUPING SETS ((a,b,c),
                        (a,b),
                        (a,c),
                        (b,c),
                        (a),
                        (b),
                        (c),
                        ());
```

També podem combinar CUBE i ROLLUP per obtenir els totals que ens interessin.

```
GROUP BY CUBE(a,b), ROLLUP(c,d)
```

és equivalent a:

```
GROUP BY GROUPING SETS ((a,b,c,d),
                        (a,b,c),
                        (a,b),
                        (a,c,d),
                        (a,c),
                        (a),
                        (b,c,d),
                        (b,c),
```

```
                    (b),
                    (c,d),
                    (c),
                    ())
```

L'estàndard permet combinar CUBE, ROLLUP i GROUPING SETS de qualsevol manera per obtenir el resultat desitjat.

Utilitzar CUBE, ROLLUP i GROUPING SETS, a més de facilitar l'escriptura de les consultes, també millora el rendiment del sistema perquè facilita informació extra a l'optimitzador de consultes sobre el que es pretén fer.

6. Disseny físic

Una vegada ja tenim les relacions que componen la nostra base de dades i també sabem quin tipus de consultes volem executar, només falta fer el disseny físic tenint en ment que cal aconseguir un bon temps de resposta a les consultes.

6.1. Pla i tècniques bàsiques d'accés

Abans de veure quines eines específiques tenim des del punt de vista físic per millorar el rendiment del sistema, penseu com seria el pla d'accés d'una consulta multidimensional (aquest pla pot tenir petites variacions segons la SGBD):

1) Avaluar les condicions sobre cadascuna de les Dimensions per obtenir un conjunt d'identificadors.

SGBD

Cal assegurar-se que l'SGBD reconeix l'esquema amb forma d'estrella i utilitza aquest pla d'accés.

2) Combinar (fer el producte cartesià) els identificadors de totes les Dimensions per obtenir els identificadors del Fet que ens interessen.

3) Buscar el Fet per obtenir els valors (mesuraments) que volíem.

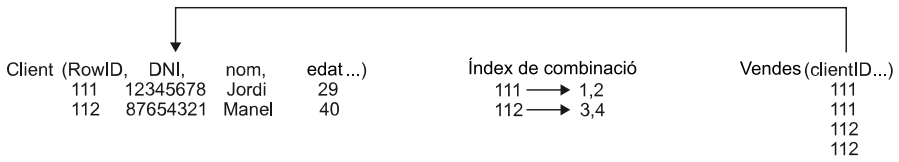
4) Ordenar els resultats.

5) Agrupar i operar mesuraments.

En el segon pas del pla d'accés que acabem de veure, es fa el producte cartesià de tots els identificadors seleccionats a les Dimensions. Tot i que aquesta operació és molt costosa, no ho és tant com fer de manera seqüencial la combinació de la taula de Fet (sens dubte, la més gran de totes) amb cadascuna de les taules de Dimensió.

La pregunta que ens podem fer ara és com es pot abaratir encara més aquest segon pas. Podem evitar el producte cartesià? La resposta és que sí, mitjançant índexs de combinació (*join indices*). Un índex de combinació és aquell definit sobre una clau forana, de manera que té els valors d'una taula i apunta a l'altra. Com podeu veure a la figura següent, en certa manera, l'índex conté el resultat de la combinació precalculat.

Figura 39



Quan definim una clau primària en una taula, el SGBD defineix un índex B^+-arbre sobre els atributs corresponents. Observeu que en una taula de Fet, la clau primària està formada per claus foranes cap a les taules de Dimensió. Per tant, realment parlem d'un índex de combinació. Amb els identificadors d'una Dimensió, podem recórrer l'índex fins a tenir la branca o branques que ens interessen: no caldrà fer el producte cartesià de totes les Dimensions.

Figura 40

El problema que té aquest tipus d'índex és que cal que les condicions de la consulta afectin les Dimensions corresponents als primers atributs de la clau primària. Recordeu que en un índex B^+-arbre l'ordre dels atributs és rellevant. No és el mateix construir un índex sobre la data i els clients, que sobre els clients i la data. Mentre que per utilitzar el primer cal haver fixat la data, per utilitzar el segon s'ha d'haver fixat el client. Sempre entrem en l'índex pel valor corresponent al primer atribut i comprovem els valors de la resta dels atributs de manera seqüencial. Per tant, per utilitzar un índex B^+-arbre, cal que l'usuari hagi fixat el valor o els valors del primer atribut. Com que moltes consultes multidimensionals es fan restringint la data, aquesta seria una bona elecció per al primer atribut d'un índex de combinació.

En una taula, a més de l'índex de la clau primària, podem definir tants índexs com vulguem, però definir-ne un per a cada combinació de Dimensions que pugui arribar a demanar l'usuari és possible que signifiqui sobrecarregar el sistema amb la gestió d'índexs inútils.

Contingut complementari

Definició dels índexs

Definir malament els índexs pot significar que una consulta tardi hores en executar-se.

Els índexs B^+-arbre són especialment útils per fer consultes simples (sense agrupacions, ni agregacions, ni moltes combinacions), i funcionen millor com més gran és la selectivitat de l'atribut (com menys valors repetits té). El problema principal en aquest cas és que els atributs de les consultes multidimensionals no solen ser gaire selectius. A més, per a taules molt grans, l'índex B^+-arbre pot ocupar massa espai. Si la taula de Fet conté menys de cinc Mesures, un índex B^+-arbre pot ocupar un 80% de la mida de la taula.

Els índexs B^+-arbre poden ser útils per resoldre algunes consultes multidimensionals, però no n'hi ha prou amb això.

Observeu, també, que definir un índex com agrupat (*cluster*) pot resultar molt profitós i no gaire costós si el primer atribut és la data. Com que les insercions es fan de manera massiva i de data en data, sempre aniran a parar al final de la taula. Si la taula està ordenada per data, les dades que vam inserir ahir han d'anar abans que els que inserim avui, que han d'anar abans que les que inserirem demà, etc. D'aquesta manera, la taula queda ordenada sense cap cost addicional.

Índex agrupat

Recordeu que un índex agrupat és aquell que no només manté ordenat l'índex, sinó també les dades a l'interior de la taula.

6.2. Índexs de mapes de bits

La millor opció (encara que no disponible en tots els SGBD) per indexar taules de Fet són els índexs de mapa de bits (*bitmap*). Un índex de mapa de bits és una matriu booleana de tantes columnes com valors tingui l'atribut utilitzat per a la indexació i tantes files com la taula indexada. Si la posició [i,j] val cert, aleshores a la fila *i*-èssima l'atribut pren el valor *j*-èssim.

Figura 41

Bolígrafs	Llapis	Portalàmines	Gomes	Fulls A4	Fulls A3	Guixos	Esborradors
1	0	0	0	0	0	0	0
0	0	1	0	0	0	0	0
0	1	0	0	0	0	0	0
0	0	0	0	0	0	0	1
0	0	0	0	1	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	1	0	0
0	0	1	0	0	0	0	0
0	0	0	0	0	0	1	0
0	1	0	0	0	0	0	0

Catalunya	Castella i Lleó	Madrid	Andalusia
1	0	0	0
1	0	0	0
0	0	0	1
0	0	1	0
0	1	0	0
1	0	0	0
0	0	0	1
0	1	0	0
1	0	0	0
1	0	0	0

Exemple de mapes de bits

A la figura superior, podeu veure dos exemples de mapes de bits per a la taula de Fet `Vendes`, un per a l'atribut `Producte.nom` i l'altre per a `Regio.nom`. Podem veure que la primera i sisena files de la taula de vendes corresponen a vendes de bolígrafs (observant la primera columna del mapa de bits esquerre), i que la quarta correspon a una venda a Madrid (observant la tercera columna del mapa de bits dret).

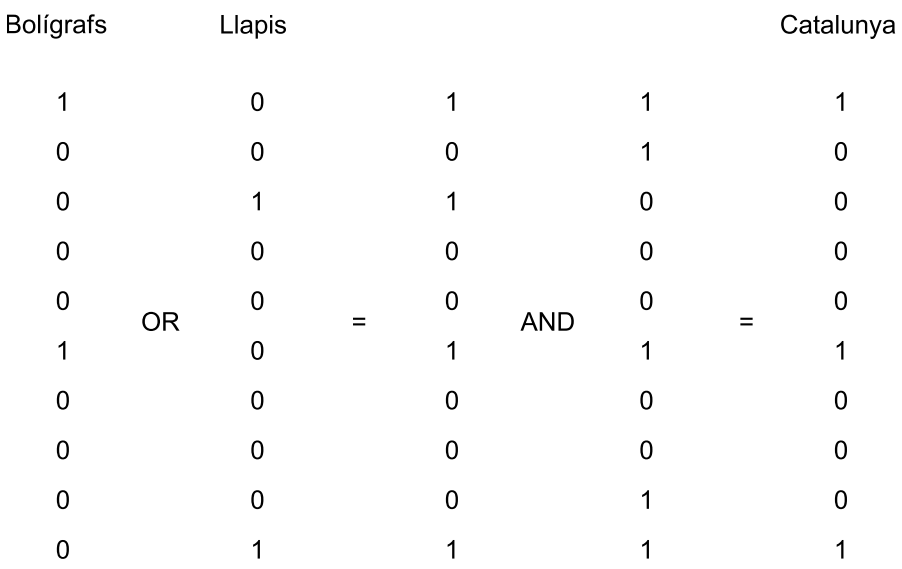
Al contrari que els B^+-arbre, els mapes de bits resulten especialment interessants per a atributs amb pocs valors possibles i una baixa selectivitat (el mateix valor repetit moltes vegades). Es tracta de l'índex ideal per utilitzar en la taula de clients amb un atribut com per exemple `sexe` (home o dona), però no té cap sentit utilitzar-lo per a `DNI`, perquè tothom tindrà un DNI diferent (obtindríem una matriu molt gran en la qual cada columna només tindria un valor cert).

Aquest tipus d'índex ocupa molt poc espai (si la taula tingués un milió de files i l'atribut vuit valors possibles, l'índex ocuparia només 1 *Mbyte*) i permet utilitzar operadors lògics de baix nivell per resoldre predicats sobre els atributs (només cal fer AND i OR sobre seqüències de bits, que són operacions molt ràpides). A més, s'obté un rendiment òptim per resoldre consultes que no necessiten accedir a les dades. Només accedint a l'índex, podem comptar quantes files compleixen o deixen de complir una certa condició (comptant zeros o uns, de manera respectiva). En els mapes de bits de la figura anterior podem veure que només tenim una venda d'esborradors i cinc vendes a Catalunya, sense conèixer el contingut de la taula de Fet.

Els mapes de bits són fàcils de construir, mantenir i utilitzar.

Una possible manera d'implementar-ho consisteix a definir un B^+-arbre sobre una certa Dimensió i guardar matrius de bits a les fulles de l'arbre que indiquin quines files de la taula de Fet apunten al valor d'aquesta taula de Dimensió corresponent a la fulla en qüestió. Us heu de fixar en la diferència que hi ha amb l'apartat anterior, en el qual definíem un sol arbre per a tota la taula de Fet. Ara tenim un arbre diferent per a cada taula de Dimensió, cadascun dels quals conté mapes de bits de la taula de Fet. Amb això, si volem fer una consulta, utilitzarem l'índex de cada Dimensió per obtenir el mapa de bits corresponent, operarem amb tots i accedirem directament a les files indicades.

Figura 42



Exemple d'ús dels mapes de bits

A la figura 42, podeu veure com utilitzaríem els mapes de bits de les Dimensions `Producte` i `Lloc` per saber quines vendes de bolígrafs i llapis s'han fet a Catalunya. Per obtenir les dades, només cal accedir a les files primera, sisena i desena.

Mapes de bits

Una variant dels mapes de bits consisteix a utilitzar un bit per a cada pàgina de disc, en lloc de cada fila. Per millorar els mapes de bits, alguns sistemes també els apliquen tècniques de compressió i codificació.

Els índexs de mapa de bits són especialment adequats per fer consultes amb moltes condicions sobre diferents atributs que tenen una selectivitat baixa. Si l'atribut sobre el qual definim l'índex té molts valors possibles, la matriu resulta molt dispersa i malbarata molt espai.

6.3. Particions horitzontals

Com ja hem dit bastantes vegades, una taula de Fet realment és molt gran. Al mateix temps, accedir-hi resulta imprescindible per resoldre qualsevol consulta. Per consegüent, hem de facilitar l'accés a aquesta taula tant com puguem. Una manera de fer-ho consisteix a dividir-la en *n* subtaules o particions, posant en cadascuna l'enèsima part de les files (és millor que les particions no se superposin, que no tinguin files en comú). Sempre podem obtenir fàcilment la taula sencera fent la unió de les particions.

Cada partició pot estar en un disc diferent o, fins i tot, en una màquina diferent, amb el corresponent guany de rendiment (aplicant tècniques de paral·lelisme). Com a efecte col·lateral, també facilitem l'escalabilitat del sistema. Quan necessitem més espai, no fa falta un disc on puguem posar totes les dades, sinó simplement un on puguem posar com a mínim una de les particions. L'inconvenient en aquest cas és la disponibilitat. N'hi haurà prou amb què una sola màquina o disc no funcioni perquè no puguem respondre la consulta.

Podem definir les particions segons els valors de qualsevol Dimensió (o fins i tot de més d'una Dimensió). Només us heu d'assegurar de què aquesta Dimensió no canviarà mai. La reestructuració de totes les particions pot generar problemes greus. A més, tingueu en compte que fer particions és especialment útil (encara que no disposem de paral·lelisme, ni tan sols de discos diferents), si una consulta només ha d'accedir a una de les particions o com a mínim no ha d'accedir a totes. Per tant, hem de fer particions segons un atribut que s'acostumi a fixar sempre en les consultes. Una altra vegada la data pot ser una bona elecció. Podem tenir una partició per a cada mes, cada setmana o cada dia, si cal.

La partició horitzontal d'una taula de Fet facilita el paral·lelisme i l'escalabilitat del sistema, alhora que redueix el temps de resposta per a les consultes que utilitzin per seleccionar els Descriptors de la Dimensió utilitzada com a criteri de partició.

6.4. Particions verticals, eines VOLAP

De la mateixa manera que podem fer una partició horitzontal de la taula del Fet, per reduir la seva mida, també podem fer-la verticalment. En aquest cas, definim diferents taules amb subconjunts dels atributs, amb el benentès que cadascun ha d'incloure la clau primària. Com que habitualment cada consulta conté només una Mesura, ho podem dur a l'extrem i definir una taula diferent per a cadascuna, com podeu veure a la figura següent.

Figura 43

Vendes1(IDTemps, IdClient, IdProducte, IdLloc, IdPromoció)

Vendes2(IDTemps, IdClient, IdProducte, IdLloc, articles)

El guany de la partició vertical és que maximitzem la proporció de dades útils respecte de les dades a les quals s'ha accedit. Imaginem que cada registre abans de la partició ocupa 28 *bytes* i, després de la partició, només n'ocupa 24. Aleshores, cada vegada que accedim a una pàgina de disc, obtenim aproximadament un 16% més de valors de la Mesura que ens interessa. Observem, no obstant això, que pràcticament necessitem el doble d'espai per tenir `Vendes1` i `Vendes2` que el que es necessitava abans per tenir només `Vendes`, sense particions de cap tipus.

Per millorar el rendiment del sistema, podem definir particions verticals de la taula de Fet. Cadascuna de les particions contindrà la clau primària i una o més Mesures.

La millora en aplicar aquesta tècnica seria molt més important si no s'hagués de replicar la clau primària en cadascuna de les taules. En aquest cas, un registre ocuparia el mínim indispensable per guardar un valor, i tot el contingut de la pàgina de disc a la qual s'ha accedit ens seria útil. Doncs bé, hi ha eines OLAP que fan exactament això. Aquestes eines reben el nom de VOLAP.[10]

(10) *Vertical OLAP.*

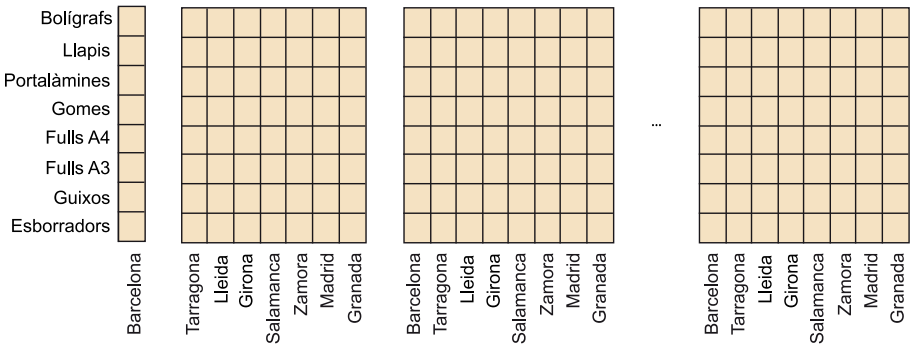
Les eines VOLAP ja no són relacionals. Es tracta de SGBD específics per a anàlisis multidimensionals que guarden els mesuraments d'una mateixa Mesura totes seguides sense els valors de la clau primària, i que utilitzen diferents tipus d'índexs per localitzar-los. Aquests sistemes poden obtenir un temps de resposta més de deu vegades millor que un SGBD relacional, i utilitzen fins a cent vegades menys espai (apliquen mecanismes de compressió). El problema que tenen és que es tracta de solucions totalment propietàries, no estandarditzades, i que es basen en el supòsit que només volem veure les Mesures d'una en una.

Si les consultes només demanen valors d'una sola Mesura, podem utilitzar eines VOLAP.

6.5. Matrius *n*-dimensionals, eines MOLAP i HOLAP

Hi ha gent que creu que les bases de dades relacionals no són adequades per a l'anàlisi multidimensional i que utilitzar-les és artificiós. De fet, van ser concebudes per a un món transaccional. Si el que volem consultar són cubs, per què emmagatzemem taules? La resposta són les eines *MOLAP* (*multidimensional OLAP*) o multidimensionals pures.

Figura 44

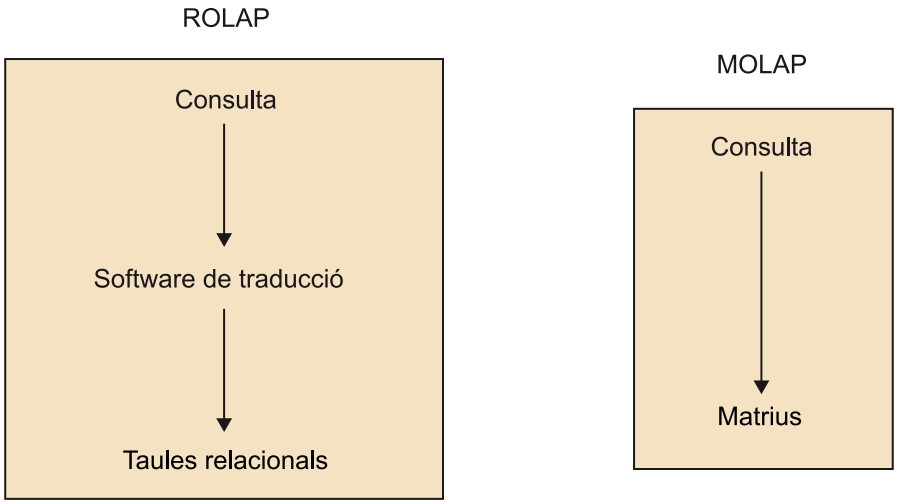


Les eines MOLAP emmagatzemen matrius, com la que teniu a la figura 44, i implementen sistemes d'indexació especials per accedir-hi. Ara ja no parlem de claus primàries, ni de claus foranes. Cada element de la matriu conté només les Mesures. Aquestes matrius tenen exactament la mateixa estructura que els cubs que volem acabar visualitzant. És més, les matrius es defineixen en funció dels cubs que seran consultats amb més freqüència. És a dir, atès el conjunt

de consultes crítiques (les executades més sovint i que demanen un temps de resposta més baix), l'eina MOLAP cerca la manera d'emmagatzemar les dades per minimitzar el temps de resposta per a aquestes consultes.

Les eines MOLAP utilitzen com a sistema d'emmagatzematge matrius *n*-dimensionals, en lloc de taules relacionals.

Figura 45



A la figura 45 podeu veure la diferència de filosofies entre una implementació ROLAP i una MOLAP. Mentre que la ROLAP necessita una traducció de la consulta a SQL, en una implementació MOLAP la consulta es resol directament sobre les dades (no hi ha cap pas intermedi).

Atès que no emmagatzemen claus primàries, pot semblar que les eines MOLAP fan el mateix que les VOLAP. Això no és cert. La diferència entre una eina MOLAP i una VOLAP és que la MOLAP emmagatzema les dades segons les consultes que s'esperen, mentre que la VOLAP, no. En aquest sentit, una eina VOLAP està més prop d'una ROLAP que d'una MOLAP.

Afavorir unes consultes més que d'altres és una mica més natural del que sembla. Per molt que vulguem emmagatzemar matrius *n*-dimensionals, el disc només té dues Dimensions (cilindres i sectors) i la memòria RAM, simplement una. Per tant, encara que una eina MOLAP intentés gestionar totes les Dimensions de manera similar, no ho podria fer. N'hi hauria alguna a la que s'accediria de manera més eficient que a les altres, perquè les dades estarien físicament agrupades. Per exemple, podríem tenir la informació dels dotze mesos de l'any dins del mateix sector i cilindre, de manera que accediríem a tots sense necessitat d'esperar tota la rotació de disc, ni moure el braç per canviar de cilindre. Per contra, si tenim un mes a cada sector, haurem d'esperar que el disc giri completament per tenir-los tots.

Aquest sistema d'emmagatzematge amb matrius dependents de les consultes és molt eficient, però massa rígid. Penseu què passa a la figura 44 si volem afegir una nova data. No hi ha cap problema: simplement afegim a la dreta una nova matriu de 8 × 8. No obstant això, què passa si volem afegir una nova ciutat? Cal reorganitzar tota la matriu per tenir lloc per posar els nous elements allà on els toca. Per exemple, hauríem d'obrir un forat entre les dades de Granada i Barcelona per posar els nous elements al mig. A més, afegir Dimensions a una eina MOLAP multiplica l'espai de disc utilitzat i resulta molt més complex que en una eina ROLAP, en la qual simplement cal afegir un nou atribut a la clau de la taula del Fet.

A més del problema de la rigidesa, trobem el problema de les consultes no considerades crítiques o simplement imprevistes. Com que el sistema no les ha tingut en compte per emmagatzemar les dades, és bastant probable que tardi molt a resoldre-les.

Les eines MOLAP funcionen especialment bé quan tenim poques Dimensions i molt estables (que mai no canvien). Donen molt bon resultat respecte al temps de resposta, però generalment un mal resultat pel que fa a l'emmagatzematge, especialment per a cubs amb una dispersió gran. Això es deu al fet que es guarda el cub sencer, tant les cel·les plenes com les buides, per facilitar la indexació i l'accés. Per resoldre l'excés d'espai utilitzat, solen utilitzar tècniques de compressió de dades.

El principal avantatge d'una eina MOLAP és la seva rapidesa per consultar les dades. Els seus punts febles són la gestió de grans volums de dades i la rigidesa davant dels canvis. Funcionen bé per a magatzems de dades departamentals relativament petits i estables.

Eines MOLAP

Algunes eines MOLAP intenten aprofitar més l'espai aplicant tècniques de compressió.

Encara que el temps de resposta sigui millor amb les eines MOLAP, les eines ROLAP s'imposen al mercat més per la mateixa implantació i el nivell de desenvolupament dels sistemes relacionals que perquè siguin realment adequats per a tasques multidimensionals. L'estandardització del llenguatge SQL és un punt a favor molt important d'aquest tipus d'eines, ja que facilita la definició de traductors de consultes multidimensionals independents de la SGBD. No obstant això, a més de l'estandardització del llenguatge, també és cert que les eines ROLAP demostren una gran robustesa i flexibilitat per tractar grans volums de dades com els de les taules de Fet. Observeu també que amb la implementació sobre un SGBD relacional no tenim problemes amb la dispersió dels cubs. Només hi ha files per a les cel·les que contenen dades. A més, les eines MOLAP baixen molt rendiment quan es fan consultes imprevistes.

Per aprofitar el millor de les eines MOLAP i les ROLAP, també hi ha eines *HOLAP* (*hybrid OLAP*). Les eines MOLAP són millors per a cubs densos. Les eines ROLAP són millors per a cubs dispersos. Les eines HOLAP identifiquen regions

denses i disperses i les emmagatzemen segons una tècnica o una altra. Quan arriba una consulta, la desfan segons la regió del cub involucrat i buscaran les dades a un tipus de servidor o a un altre.

Les eines HOLAP barregen el millor de les eines ROLAP i MOLAP.

6.6. Tècniques de preagregació

Com hem vist fins ara, els diferents tipus d'eines multidimensionales aprofiten les diferents característiques de les consultes per millorar el temps de resposta. Les eines ROLAP aprofiten especialment la divisió entre Fets i Dimensions; les VOLAP aprofiten que el més habitual és demanar una sola Mesura; les MOLAP, la repetició i previsió de consultes, etc. Doncs bé, hi ha una característica de l'anàlisi multidimensional que aprofiten totes aquestes: la freqüència d'ús de les operacions *roll-up* i *drill-down*. Sabem que l'usuari demanarà les dades a diferents granularitats. Una manera de millorar el temps de resposta consisteix a tenir calculat el resultat d'aquestes consultes abans que ho demani.

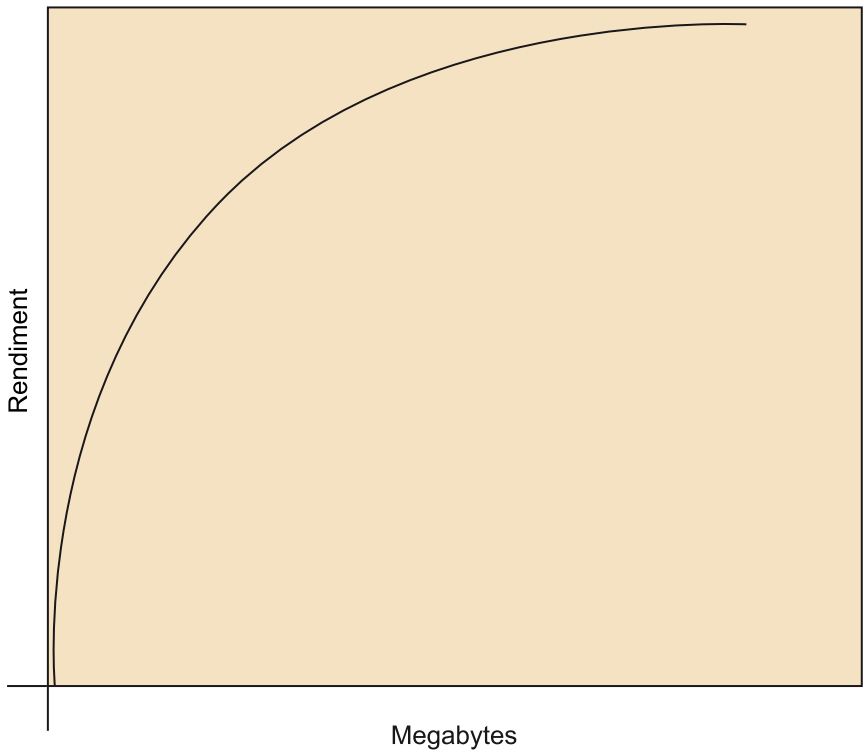
De manera independent del tipus d'eina que utilitzem, sempre es poden aplicar tècniques de preagregació per millorar el temps de resposta davant de les operacions *roll-up* i *drill-down*. La preagregació és l'eina més potent per millorar el temps de resposta d'una aplicació multidimensional.

Recordeu que a l'apartat «Components del model multidimensional» ja hem vist que cada Fet conté un reticle de Cel·les relacionades per agregacions. Malgrat que en el disseny conceptual només representem algunes (les més importants) d'aquestes Cel·les, l'usuari les voldrà veure totes en un moment o en un altre. Per tant, el sistema ha de garantir que aquestes dades estan disponibles i es poden obtenir amb rapidesa.

El cost d'obtenir dades agregades no ve donat pel càlcul que s'hagi de fer, sinó per la simple obtenció de les dades que cal afegir.

La primera solució és guardar les instàncies de totes les Cel·les. Es tracta de la solució més ràpida (quant al temps de resposta) i la que malbarata més espai. Observeu que el nombre de Cel·les creix de manera exponencial respecte al nombre de Dimensions i Nivells. Simplement, un Fet amb n Dimensions amb un Nivell per a cadascuna tindrà 2^n Cel·les. Per desgràcia, això fa que en la majoria dels casos aquesta opció sigui inviable.

A l'extrem oposat, podem emmagatzemar les instàncies de la Cel·la de menys granularitat i calcular les instàncies de les altres Cel·les sota demanda. Aquesta és la solució que ocupa menys espai, però també la més lenta. El que cal fer realment és decidir quines Cel·les guardem físicament (dit d'una altra manera, materialitzades) i quins calculem només quan la consulta ho demani.



Com podeu veure a la gràfica, dedicant molt pocs *Megabytes*, millorem molt el rendiment. No obstant això, arriba un moment en el qual, per més espai que dediquem a emmagatzemar Cel·les, no millorem gens el rendiment del sistema.

Per prendre aquesta decisió, us heu d'adonar que, com més alta estigui una Cel·la en el reticle, menys dispersió hi haurà. Per exemple, n'hi ha prou que un dia hàgim venut alguna cosa perquè, quan agreguem els mesos, ja tinguem una cel·la per al mes corresponent. Si ens imaginem que només tenim dades d'aquest mes, a granularitat Dia, tenim una dispersió 31/1 (només un dels trenta-un punts de l'espai conté una Cel·la), mentre que, a granularitat Mes, tenim una dispersió 1/1 (l'únic punt de l'espai conté una Cel·la). En un cas com aquest, necessitaríem el mateix nombre de tuples per emmagatzemar les dades detallades i per emmagatzemar els agregats, i no aconseguiríem cap guany en el temps de resposta. Com més disperses siguin les dades bàsiques, més espai ocuparan de manera proporcional les dades agregades.

Abans d'emmagatzemar físicament una Cel·la, ens hem d'assegurar que cadascuna de les seves instàncies resulta de l'agregació de, com a mínim, deu instàncies de la Cel·la a partir de la qual la calculem. Si no, el petit guany de temps de resposta no compensarà la despesa d'espai extra.

L'espai que necessitaríem per emmagatzemar totes les Cel·les és d'ordres de més magnitud que la que ocupen les dades bàsiques. L'experiència diu que hem de dedicar aproximadament el mateix espai per a dades preagregades que el que ocupen les dades bàsiques.

Ja hem pres una primera decisió a l'hora de fer el disseny conceptual. Les Cel·les que hem explicitat (perquè contenen Mesures que no podem calcular a partir de les Mesures de les altres Cel·les o perquè l'usuari les considera especialment importants) són les primeres candidates per emmagatzemar. Ara només cal veure quina de les altres volem emmagatzemar al costat d'aquestes. Per exemple, si emmagatzemem vendes mensuals (que denotarem `Vendes(Client,Producte,Poblacio,Mes))`, ja no cal que consultem les vendes diàries `(Vendes(Client,Producte,Poblacio,Dia))` per calcular les vendes anuals `(Vendes(Client,Producte,Poblacio,Any))`, perquè és molt més econòmic fer-ho a partir de les mensuals.

Decidir quines dades cal preagregar per obtenir un temps de resposta pròxim al qual obtindríem preagregant-les totes, però dedicant un espai de disc i un temps d'actualització raonables, dependrà de molts factors (per exemple, el maquinari disponible, les característiques de la xarxa i del programari, el nombre d'usuaris, etc.). A més, mai no trobarem el conjunt perfecte de Cel·les que cal preagregar, perquè les consultes dels usuaris evolucionen amb el pas del temps.

El problema de triar quines Cel·les materialitzem no resulta gens trivial (tenint en compte la seva complexitat computacional, es tracta d'un problema NP-complet). No obstant això, una bona solució és ordenar-les segons la seva utilitat. Les Cel·les més útils són les que serveixen per resoldre les consultes més costoses i comunes. Una Cel·la és útil tant si la demana l'usuari amb molta freqüència, com si serveix per obtenir fàcilment la que demana sovint l'usuari. Una Cel·la no és gaire útil si ja hem decidit emmagatzemar les instàncies d'una altra Cel·la a partir de la qual ja podem obtenir-la fàcilment. Una vegada feta aquesta ordenació, si sabem l'espai que ocupa cada Cel·la i el que tenim disponible, apliquem un algorisme voraç (*greedy*) i emmagatzemem tantes Cel·les com puguem, seguint l'ordre d'utilitat fixat anteriorment. Alguns sistemes ja fan l'elecció de manera automàtica, quan l'administrador indica l'espai de disc que li vol dedicar.

A l'hora de decidir l'espai que volem dedicar a guardar dades preagregades, he- mos de tenir en compte que serà necessari actualitzar les dades agregades cada vegada que modifiquem la Cel·la atòmica. Observeu que això pot fer créixer molt la finestra d'actualització. Per tant, per decidir l'espai dedicat, no només hem de mirar el preu del disc, sinó el temps necessari per mantenir actualitzades aquestes dades preagregades.

La millor estratègia per decidir quines dades tenim preagregades és l'assaig-error.

7. Beneficis d'una presentació adequada de dades

Una anàlisi adequada de les dades existents en una empresa o organització pot facilitar enormement la comprensió dels negocis i mercats, i és de gran ajuda per a la presa de les decisions empresarials correctes. Per dur a terme aquesta anàlisi, és necessària la utilització de les eines precises que incrementin l'eficiència organitzacional i la seva efectivitat, agilitant el flux de dades dins de l'organització.

La visualització de dades en condicions òptimes fomenta l'intercanvi d'informació i millora la comunicació. A més, una altra particularitat de les visualitzacions és que transmeten la informació d'una manera universal, aconseguint que resulti molt senzill explicar alguna cosa i que a més es pugui compartir amb altres membres de l'equip o amb integrants de l'organització en diferents àmbits.

Sens dubte, la cultura organitzativa i el nivell de maduresa en el qual es trobi una organització determinaran quins instruments s'implantaran i quan:

- Si no hi ha una cultura del mesurament, el projecte ha de fonamentar-se en el *reporting* operacional, en el *reporting* tàctic i en anàlisis multidimensionals controlades.

- Si la cultura del mesurament està present però no arrelada, s'ha de donar pas als quadres de comandament tàctics, fomentant que els comandaments intermedis comencin a utilitzar mètriques de control.

- Si hi ha una cultura del mesurament molt arrelada, el projecte s'ha de centrar en madurar el *reporting*, l'anàlisi i els quadres de comandament estratègics, i començar amb el descobriment i la mineria de dades.

Dins del projecte, també ens hem de plantejar la definició i consolidació de la «veritat corporativa»: cal definir uns informes estàndard que siguin els oficials i uns camins d'anàlisi bàsics sobre aquests informes, de manera que tothom sàpiga on i com s'ha d'accedir a la informació.

Un altre punt que cal tenir en compte és la uniformitat de l'organització semàntica de la informació.

> Per exemple, si es pregunta a diversos usuaris què entenen per marge de l'organització, i obtenim diferents respostes (un ho defineix abans d'aplicar impostos, un altre després, i un altre indica que cal descomptar els abonaments), aleshores haurem de crear un document de semàntica comuna abans de començar el projecte. Aquest document deurà explicar «què s'entén per...» i, a més, cal assegurar que tots els implicats utilitzin la mateixa semàntica.

Amb les dades d'un sistema BI, és possible generar informes globals o per seccions, crear escenaris respecte a una decisió, fer pronòstics o anàlisis multidimensionals, generar i processar dades, etc.

A continuació, s'enumeren alguns exemples de les àrees més comunes en les quals s'utilitzen les solucions d'intel·ligència de negoci:

- Vendes: anàlisi de vendes; detecció de clients importants; anàlisi de productes, línies, mercats; pronòstics i projeccions.

- Màrqueting: segmentació i anàlisi de clients; seguiment a nous productes.

- Finances: anàlisi de despeses i ingressos; rotació de cartera; raons financeres.

- Manufactura: productivitat en línia; anàlisi de rebuig; anàlisi de qualitat; rotació d'inventaris i parts crítiques.

- Embarcaments: seguiment d'embarcaments; motius pels quals es perden comandes.

Trobem una gran diversitat d'enfocaments i pràctiques que varien d'un país a un altre, però és universalment acceptat el fet que una de les millors tècniques per fer comprensibles les dades és la representació dels números mitjançant imatges. Això pot fer molt més fàcil apreciar un patró o exposar certs patrons que, d'una altra manera, podrien quedar ocults.

La informació a la qual accedeixen els usuaris resulta del tractament i presentació adequats de les dades per a la presa de decisions en tres nivells:

1) Nivell operatiu: permet als usuaris d'aquest nivell, que en la seva rutina diària gestionin informació, que la rebin d'una manera oportuna, exacta i adequada.

Manipulen principalment eines d'informes o fulls de càlcul amb formats fixos que estan en constant actualització.

> Posem l'exemple d'un supervisor de vendes el suport principal del qual, per fer satisfactòriament la seva feina, és un full de càlcul, amb el qual monitoritzaria el fet que es compleixin les quotes de vendes dels venedors a càrrec seu. Una de les columnes tindria informació fixa (quota de vendes) i en la següent s'anotarien les dades diàries de vendes (en termes monetaris) de cadascun dels seus venedors, i amb aquestes dades podria fer una anàlisi i prendre decisions respecte als objectius que compleix cada treballador o, en el seu cas, les seves deficiències.

2) Nivell tàctic: aquest nivell permet analitzar i consultar informació en un nivell mitjà, mitjançant la utilització d'eines informàtiques, sense intervenció de tercers.

Suposem, per exemple, que a un gerent li arriba un informe imprès, o preimprès, on indica que les vendes en algun producte o servei s'han elevat considerablement.

Una eina d'aquest nivell ha de permetre dur a terme una anàlisi que marqui les tendències i variables d'un possible increment. Gràcies a això, el gerent pot adonar-se que l'increment de demanda es deu a productes, clients o estratègies promocionals. A més, aquesta eina pot determinar si és cíclic (i passa en un període definit i la seva tendència augmenta en una època determinada) o si aquests increments són puntuals.

A partir de l'anàlisi, el gerent pot dictar estratègies per augmentar l'impacte positiu o reduir l'impacte negatiu, segons el cas.

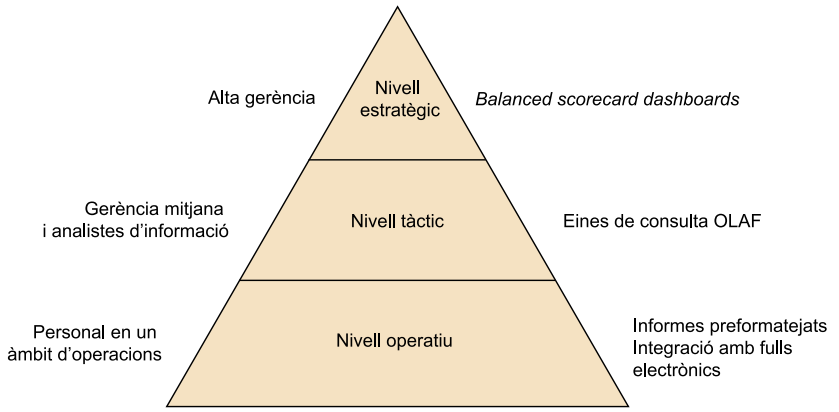
3) Nivell estratègic: permet monitorar i analitzar les tendències, patrons, metes i objectius estratègics de l'alta direcció.

En aquest nivell, és molt comú trobar quadres de comandament (*balanced scorecard*, terme introduït per Robert Kaplan i David Norton). Té concordança amb normes internacionals d'excel·lència en la gestió i assegurament de la qualitat.

És un esquema amb múltiples Dimensions per descriure, implementar i administrar una estratègia en qualsevol nivell, mitjançant la relació d'objectius, iniciatives i mesuraments a les estratègies organitzacionals. Condueixen a l'empresa a estratègies definides que es converteixen en plans d'acció.

A la figura 46, es mostra, per a cada nivell de la presa de decisions, les herramientas que s'utilitzen i els usuaris que requereixen la informació.

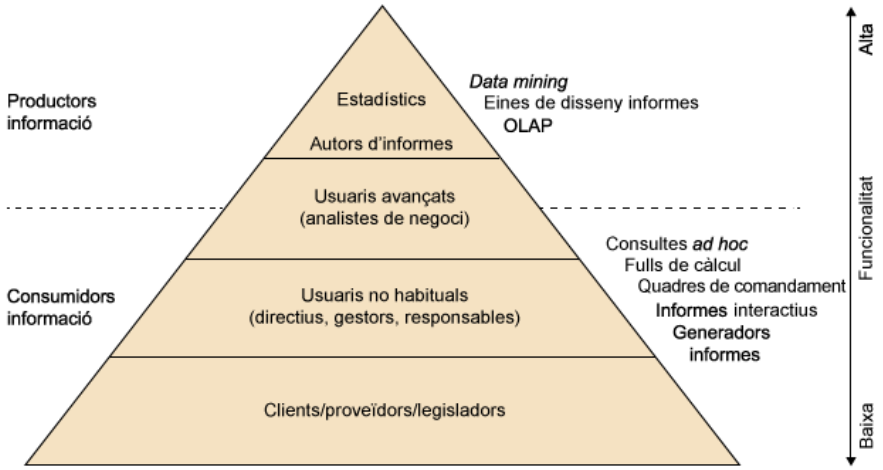
Figura 46. Eines, usuaris i nivells de presa de decisions



Font: www.gestiopolis.com

Existeixen múltiples eines per a cada nivell i se seleccionen a partir de les necessitats de cada organització. A la figura 47, veiem els perfils d'usuari de cadascuna, així com el seu nivell de funcionalitat:

Figura 47. Eines i perfils d'usuaris



Font: Cano, J. L.. *Business Intelligence*: *Competir con información*.

8. Consideracions per a la presentació de dades (riscos)

Les organitzacions són conscients dels problemes que els ocasiona la baixa qualitat de les dades, però no sempre es contemplen els inconvenients d'una presentació de les dades inadequada.

En l'enquesta *Data Quality and the Bottom Line* (Eckerson, 2002), quan se'ls preguntava quins eren aquests problemes, van afirmar:

- Temps extra per reconciliar les dades (87%).
- Pèrdua de credibilitat en el sistema (81%).
- Insatisfacció de clients (67%).
- Retards en el desenvolupament de nous sistemes (64%).
- Pèrdues d'ingressos (54%).
- Problemes de conformitat (38%).
- Etc.

I, quan se'ls preguntava pels beneficis que aporta l'adequada visualització de dades, afirmaven:

- Simple versió de la veritat (19%).
- Increments en la satisfacció dels clients (19%).
- Major confiança en els sistemes d'anàlisi (17%).
- Reducció de costos (13%).
- Menys temps per reconciliar les dades (12%).
- Increment d'ingressos (9%).
- Etc.

En les organitzacions, ens trobem amb un ecosistema de tecnologies que són vitals per al seu funcionament normal i que manipula dades complexes i dinàmiques. Moltes vegades, aquestes tecnologies són gestionades de manera individual, sense abordar una gestió corporativa conjunta, que sens dubte resultaria més eficient.

Vegem els principals avantatges que aporten les eines de visualització i anàlisi:

- Automatització de grans volums de dades: treballar manualment les dades seria inexacte i fins i tot arriscat.

- Integració: destaquen, sobretot, les seves funcions d'integració amb altres aplicacions específiques que utilitzi l'empresa.

- Accessibilitat: permet als usuaris treballar en qualsevol moment, des de qualsevol lloc i dispositiu.

- Usabilitat: interfície intuïtiva, interactiva, integrada i unificada.

- Alt rendiment en el processament de dades, tant en paral·lel com amb escalabilitat.

- Disminueix el temps d'elaboració, de manera que l'alta gerència guanyi temps per a l'anàlisi.

- Informació rellevant actualitzada.

Si es disposa de les eines adequades amb el personal inadequat o poc capacitat, l'empresa corre el risc de fracassar en el propòsit de publicar adequadament les dades, pel mal ús de la tecnologia.

Existeixen també altres circumstàncies tècniques que no contribueixen a la correcta visualització de dades, i que convé corregir o evitar sempre que sigui possible:

- La tecnologia evoluciona ràpidament. L'evolució ràpida de les tecnologies porta a introduir noves eines a l'empresa, amb la qual cosa es corre el risc que aquestes no siguin utilitzades o s'emprin de manera inadequada fins a ser compreses i dominades pels usuaris.

- Accés erroni de les dades.

- El fet que l'empresa tingui múltiples eines de BI, amb dades poc clares i objectius incompatibles amb l'organització, indueix a tenir diferents conclusions de les mateixes dades.

- No arriba a abastar tot el conjunt de processos de la intel·ligència de negoci de l'organització.

- Instal·lació complexa i problemes de rendiment.

- Davant d'un conjunt de problemes, veure solucions aïllades per departament i no com a parts d'un mateix sistema.

- Es desconeix l'impacte que la mala interpretació de les dades, escassa informació o excés d'informació poden ocasionar.

- No existeixen antecedents d'anàlisis, o els seus processos no estan degudament estandarditzats.

Les representacions han d'il·lustrar les tendències i les relacions de manera ràpida i senzilla. Son una manera eficient de transmetre la informació des de la base de dades al cap del lector.

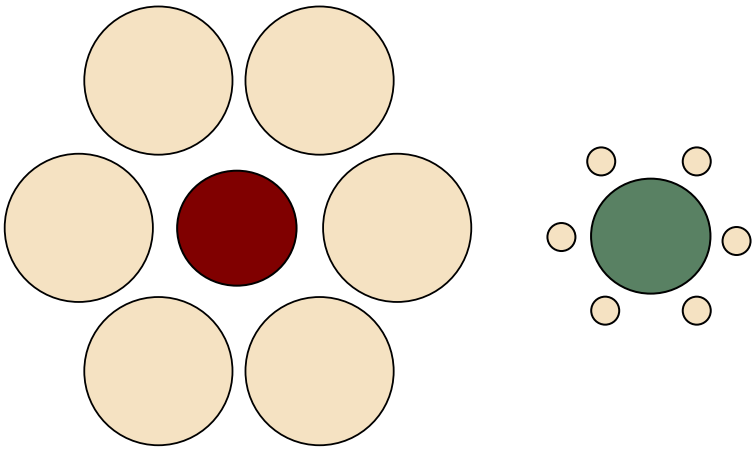
No obstant això, cal anar amb compte: una mala representació de la informació pot resultar enganyosa. Hi ha moltes maneres de proporcionar informació enganyosa, ja sigui deliberadament o, com passa de manera més freqüent, inintencionadament.

Hi ha d'haver un equilibri entre disseny i funció, ja que les representacions complicades, sovint, no aconsegueixen fer-se entendre. Com que la interpretació de gràfics pot ser complicada, no cal forçar els lectors a haver de desxifrar el missatge.

La nostra capacitat de fer observacions visuals de manera ràpida i fàcil es basa en la capacitat del cervell per percebre regularitats i irregularitats. Gran part d'aquesta capacitat funciona de manera inconscient, ja que la comparació es produeix gairebé abans de començar a pensar-hi.

Per exemple: quin dels cercles és més gran, el del centre del diagrama de l'esquerra o el del centre del diagrama de la dreta? Els cercles del centre de cada diagrama són de la mateixa mida.

Figura 48. *The Ebbinghaus Illusion*



Font: H. Ebbinghaus, 1850-1909.

Els malentesos i males interpretacions també poden ser resultat de diferents tradicions culturals. Els colors, per exemple, poden tenir diferent interpretació a diferents parts del món.

L'**experiència** també juga un paper en com es perceben els gràfics. Convé conèixer els destinataris de les dades, les seves habilitats, experiències i les seves possibles diferències. No s'ha d'assumir que tothom sap el mateix sobre un tema concret.

A continuació, s'exposen alguns aspectes que cal considerar per maximitzar l'eficiència d'un gràfic:

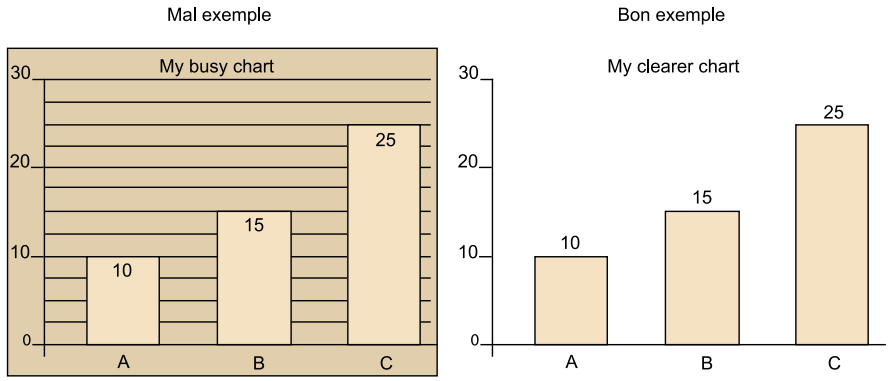
1) Les dades han d'ocupar un **lloc** central. Els components de suport han de:

- Ser exposats, només si convé. Títols d'eixos, llegendes i etiquetes de dades poden ser essencials per a la correcta comprensió del gràfic o no ser en absolut necessaris, depenent de la naturalesa de les dades.

- Ser subtils. Utilitza línies més clares per als eixos i línies de divisió per als components de dades. Els elements decoratius no han de distreure l'atenció del lector.

Vegem un exemple, a continuació, a la figura 49:

Figura 49. Exemples de diferents ubicacions de la data



Tots els components tenen un màxim impacte. El resultat és un gràfic sobrecarregat i difícil de llegir, tot i mostrar només tres valors.

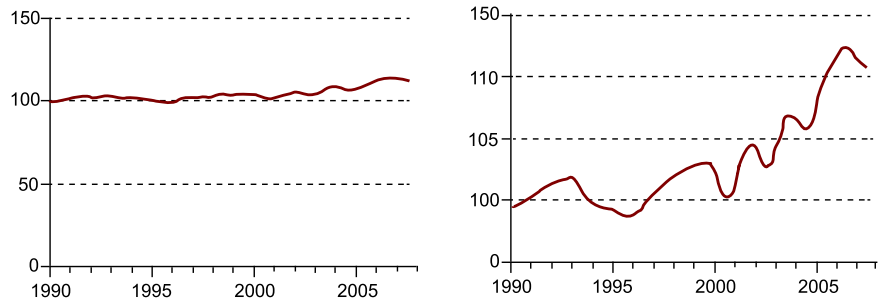
Aquest gràfic és molt més fàcil de llegir. El mínim ús de components de suport assegura que les dades ocupen el lloc central.

Font: Nacions Unides. Cómo hacer comprensibles los datos.

2) A l'hora de dissenyar un gràfic, és possible ajustar les escales per transmetre millor el missatge.

Per exemple: els dos gràfics de línies que es mostren a la figura 50 presenten les mateixes dades, però proporcionen imatges molt diferents:

Figura 50. Exemples de diferents escales de dades



Aquest gràfic mostra una tendència general d'estabilitat durant uns 10 anys, seguida d'un creixement moderat.

Reduint l'escala a l'eix Y, es poden ressaltar els canvis. Apareixen petits increments i decrements en els primers 10 anys i el creixement al final del període resulta més abrupte.

Font: Nacions Unides. *Cómo hacer comprensibles los datos.*

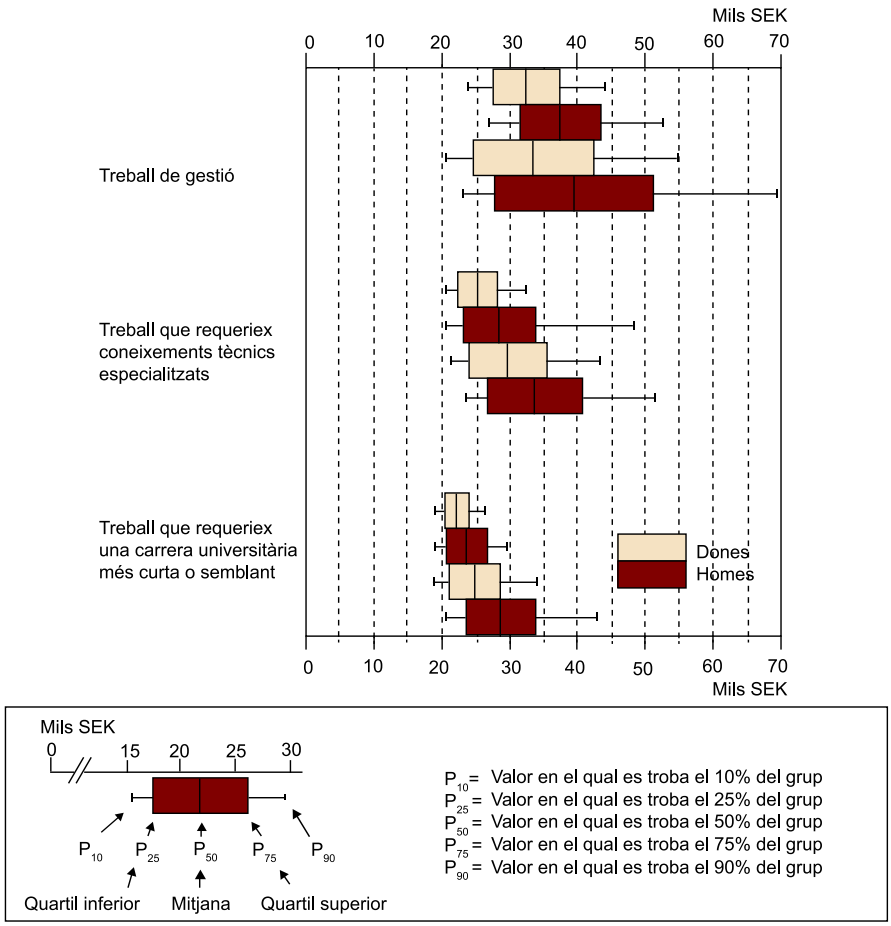
3) Les dades poden contenir diversos missatges per ressaltar en un gràfic. Als gràfics, igual que a tots els altres elements d'una publicació, se'ls pot assignar una càrrega cognitiva.

La **càrrega cognitiva** significa bàsicament quant s'ha d'esforçar el lector per entendre el que s'està intentant comunicar.

Un gràfic amb una elevada càrrega cognitiva serà difícil d'entendre i recordar, i el seu missatge resultarà difícil de comunicar. Un gràfic amb una reduïda càrrega cognitiva s'entén fent un cop d'ull, i el seu missatge resulta obvi. La majoria de les directrius sobre el disseny de gràfics eficaços estan dedicades a mantenir una càrrega cognitiva baixa.

A continuació, a la figura 51, es mostra un bon exemple de gràfic amb càrrega cognitiva alta:

Figura 51. Exemples de càrrega cognitiva d'un gràfic



Font: Nacions Unides. *Cómo hacer comprensibles los datos.*

Per resumir, les visualitzacions mal creades poden tenir efectes molt negatius per al negoci, arribant a:

- Confondre els usuaris.
- Dificultar la comprensió de les dades.
- Complicar el processament de les dades diàries.
- Fer perdre la confiança en els sistemes d'intel·ligència de negoci.

9. Formats de presentació

És l'àrea on s'han produït més avanços en els últims anys. No obstant això, la proliferació de solucions màgiques i la seva aplicació conjuntural per solucionar aspectes puntuals ha portat, a vegades, a una situació de desànim a l'organització respecte als beneficis d'una solució BI. Sense entrar a detallar les múltiples solucions que ofereix el mercat, a continuació, s'identifiquen els models de funcionalitat o eines bàsiques (cada producte de mercat integra, combina, potència, adapta i personalitza aquestes funcions).

Segons el procés que abracen i l'àmbit al qual s'orienten, les eines de *business intelligence* es poden classificar en:

- **Informes** (*reporting*): la utilitat d'aquestes eines és generar documents i informes amb un alt nivell de detall. Permeten l'obtenció d'informació actualitzada de manera molt ràpida i àgil, i la generació automàtica d'alarmes a partir d'uns criteris programats amb anterioritat.

- **Anàlisis OLAP** (*on-line analytical processing*): generació d'anàlisis molt completes amb un enfocament deductiu i amb la possibilitat d'aplicar filtres personalitzats.

- **Quadres de comandaments** (*dashboard* o *scorecard*): elaboració automàtica de diagrames i gràfics de gran poder visual, ja que permeten entendre processos i comparar dades amb un simple cop d'ull.

- **Altres**: fulls de càlcul, eines d'exploració, sistemes geogràfics, mineria de dades, etc.

9.1. Informes

El *reporting* és el precursor de la presa de decisions. A l'hora d'examinar un informe, s'ha de tenir la ment oberta per explorar el contingut i aprofundir en el que s'expressa amb xifres i percentatges. Les visualitzacions ajuden els usuaris a adonar-se de realitats que abans no eren òbvies per a ells.

El principal avantatge d'emprar un sistema visual per exposar una determinada informació és que, fins i tot quan els volums de dades són molt grans, els patrons es poden observar de manera ràpida i senzilla.

Les eines de *reporting* existeixen des de fa molt temps i són solucions madures que permeten cobrir totes les necessitats dels usuaris finals, si bé, en funció de cada fabricant, hi ha subtils diferències. Les tendències actuals per a aquestes

eines inclouen la incorporació de capacitats de visualització més grans, autoaprovisionament i funcionalitat per embeure els informes en documents (PDF, Word, PPT) o altres aplicacions de negoci. També ofereixen certa interacció, en informes que contenen taules de dades que poden ser enllaçades amb altres informes. Això implica que els usuaris puguin observar de manera resumida les dades dins d'una taula de fàcil comprensió i, llavors, si els interessa, accedir a la seva informació detallada.

S'entén per **plataforma de *reporting*** aquelles solucions que permeten dissenyar i gestionar (distribuir, planificar i administrar) informes en el context d'una organització o en una de les seves àrees.

Un informe permet respondre principalment a la pregunta de: què va passar? I està destinat als usuaris de negoci que tenen la necessitat de conèixer la informació consolidada i agregada per a la presa de decisions.

Finalment, no podem oblidar que els informes requereixen un manteniment, en el cas que canviïn els requisits i les dades, i normalment necessitarem un control detallat de l'aspecte.

9.2. Anàlisis OLAP

Les eines OLAP, que s'han estudiat en profunditat en els apartats anteriors d'aquest mòdul, permeten accedir a les dades de negoci de manera ràpida, consistent i interactiva, tenint en compte les diferents perspectives de negoci que les componen. De fet, la multidimensionalitat, que consisteix a presentar la informació de manera matricial, és una de les característiques més importants d'aquestes eines. Aquesta característica distingeix la capacitat d'aquestes eines respecte a altres sistemes de visualització de dades.

Una vista OLAP permet als usuaris investigar principalment la pregunta de: per què va passar? i, com a conseqüència, entendre quins factors són els determinants davant d'un resultat de negoci.

Per exemple, una disminució dels beneficis pot ser resultat d'un increment dels costos de producció o una reducció de les vendes. L'ús d'aquestes eines ens permet investigar aquest succés fins a trobar-ne la causa.

9.3. Quadres de comandament

Tant els informes com les eines OLAP proporcionen una gran quantitat d'informació als usuaris, que poden fer-les inadequades per prendre decisions ràpides. A més, aquestes eines responen a preguntes vinculades amb el passat: «què ha passat?» i «per què ha passat?», però no a preguntes vinculades amb el present o amb previsions de futur.

Per poder respondre a la pregunta «què està passant?», es necessita una altra estratègia/eina: el quadre de comandament, que, per la seva funcionalitat, es categoritza com una eina de monitorització.

Els quadres de comandament provenen del concepte francès *tableau du bord* i permeten mostrar informació consolidada a alt nivell. Es caracteritzen per l'ús d'elements visuals (gràfics, taules, alertes, etc.), per presentar una quantitat reduïda d'aspectes de negoci i per incloure elements interactius per potenciar i facilitar l'anàlisi de la informació en profunditat i la seva comprensió.

El quadre de comandament pot estar orientat a informació en el passat o gairebé en temps real, i a tota l'organització o a només un departament o procés de negoci. En el primer cas, la informació s'ha d'extreure del propi *data warehouse*. Per als quadres en temps real, utilitzarem els sistemes operacionals.

S'ha de tenir en compte que les respostes obtingudes són el resultat de consultes predefinides sobre la factoria d'informació. Normalment, aquestes consultes són creades per un programador d'aplicacions després d'haver recopilat les necessitats de client. L'usuari (client) de l'eina EIS no podrà modificar en cap moment els paràmetres o dades que configuren cadascun dels informes generats. Per la qual cosa, en el cas de necessitar un altre informe o una modificació dels existents, haurà de recórrer al programador.

Actualment, totes les solucions del mercat inclouen quadres de comandament perquè permeten entendre molt ràpidament la situació del negoci i són molt atractius visualment. Respecte a aquests, existeixen diferències en un àmbit de maduresa del producte, capacitats de visualització, connectivitat amb diferents fonts de dades, versatilitat d'interaccions i desplegament multidispositiu.

a) Quadre de comandament integral (*balanced scorecard*)

A part de comprendre el rendiment passat i actual d'una organització i les raons que hi ha darrere del seu comportament, les organitzacions també necessiten desenvolupar les seves estratègies de negoci.

Si bé és cert que qualsevol organització té com a objectiu a llarg termini la seva sostenibilitat i la generació de beneficis, cadascuna d'elles segueix diferents camins vinculats a la seva missió, visió i valors.

La direcció necessita un procediment formal per alinear les accions estratègiques, tàctiques i operacionals, i poder analitzar la seva evolució. La resposta a aquesta necessitat és el quadre de comandament integral.

Mostra, amb un sol cop d'ull, la comprensió del global de les condicions del negoci mitjançant mètriques i indicadors clau d'acompliment (KPI).

El quadre de comandament integral o *balanced scorecard* és una eina que permite analitzar una organització respecte a quatre perspectives: financera, de client, interna i d'innovació i aprenentatge. Aquesta eina va sorgir en els anys noranta com a resposta a la necessitat d'analitzar les organitzacions respecte al punt de vista financer, que estava quedant obsolet.

El fet d'usar les dades contingudes en un sistema de magatzem de dades és el que realment el transforma, i passa de ser un exercici puntual de revisió de l'estratègia a una eina d'alt valor per a les organitzacions. Proporciona diferents beneficis, com:

- Definir i aclarir l'estratègia.
- Habilitar una millor comunicació de l'estratègia.
- Alinear objectius personals i departamentals.
- Vincular el curt i llarg termini.
- En ser un sistema de control per excepció, permet detectar desviacions respecte al pla estratègic.

Quadre de comandament analític

És la base dels sistemes d'informació per a executius (*executive information system*, EIS) i permet la visualització ràpida i àgil de l'estat d'una determinada situació empresarial, la qual cosa possibilita la detecció d'anomalies i oportunitats.

S'elabora a partir dels *data mart*, d'informes resum i d'indicadors clau per a la gestió (KPI), que permetin als gestors de l'empresa analitzar els resultats de la mateixa de manera ràpida i eficaç. En la pràctica, és una eina de *query* orientada a l'obtenció i presentació d'indicadors per a la direcció (davant de l'obtenció d'informes i llistats).

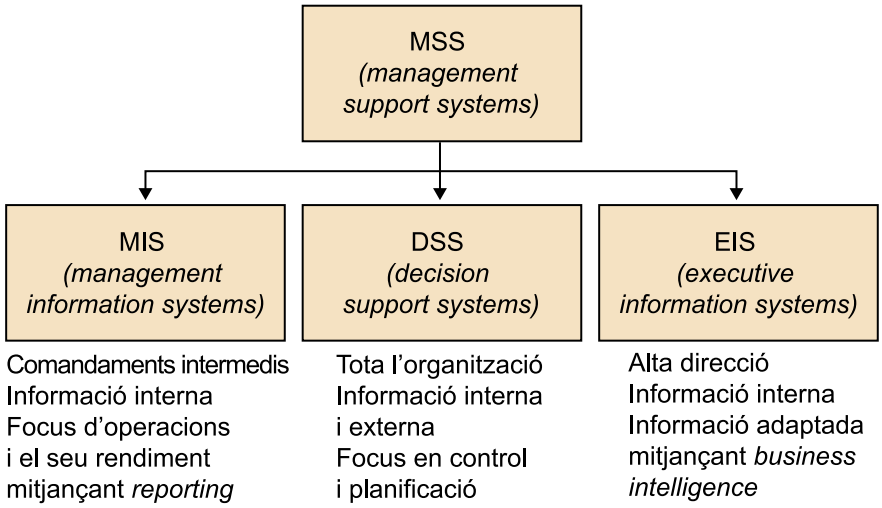
9.4. Altres

A part dels formats de presentacions vistos anteriorment, existeixen altres sistemes que ofereixen suport a la presentació de la informació.

9.4.1. Sistemes de suport a la presa de decisions (*decision support systems*, DSS)

La figura 52 resumeix els usuaris, el focus i el tipus d'informació d'aquests sistemes:

Figura 52. Estructura de sistemes de suport a la presa de decisions



Font: UOC. *Ús de la factoria d'informació corporativa.*

Si bé la discussió de les diferències entre aquests sistemes segueix essent tema d'investigació, sovint es considera que EIS és un tipus de DSS.

9.4.2. Mapes

En alguns casos, és necessària la representació sobre el territori de la informació obtinguda mitjançant eines de *business intelligence*. En aquest cas, les eines que serveixen per representar en aquest format la informació són les eines GIS o *geographic information systems*, basades en mapes. Aquestes eines afegeixen una capa de visualització sobre la qual representen els valors que obtenim de les eines BI.

Podem afirmar que un mapa és millor que milers de números. La informació geogràfica és una part essencial de moltes de les dades de la nostra organització. Les àrees geogràfiques tenen límits, noms i altra informació que permet la seva localització sobre el terreny i relacionar-hi la informació pròpia.

Els mapes són les eines més eficients per visualitzar els patrons espacials. Quan estan acuradament dissenyats i presentats, són més que simples elements decoratius en una presentació estadística, i poden ajudar les persones a identificar i ressaltar les distribucions i patrons que puguin no ser evidents en taules i gràfics.

Si «una imatge val més que mil paraules», llavors, «un mapa val més que mil números». En la nostra era visual, els mapes són una potent manera d'informar. Serveixen com a valuoses eines perquè els experts i públic en general puguin prendre decisions, així com per satisfer una creixent demanda d'informació en tots els sectors de la societat.

Existeixen molts tipus de mapes i moltes maneres de classificar-los. Una classificació especialment rellevant és la que els divideix en dos grups cartogràfics principals, en funció del tipus d'informació que aportin: cartografia base, també denominada fonamental o topogràfica, i cartografia temàtica.

- La **cartografia base** representa el tipus de mapa que originalment era l'objecte principal de la cartografia. Requereix mesures precises i es basa fonamentalment en el treball de la topografia per obtenir la informació necessària que posteriorment es plasma sobre el mapa.

- La **cartografia temàtica** se centra en la representació d'un tema concret (una variable espacial proporcionada) i pot ser de qualsevol índole: física, social, política, cultural, econòmica, etc. S'exclouen de la llista d'aquests temes possibles els purament topogràfics, que constitueixen l'objecte de la cartografia base.

Una forma habitual de treballar és crear una o diverses capes temàtiques pròpies amb les dades de la nostra organització, que podrem oferir als usuaris sobre cartografies base pròpies o disponibles de manera pública.

9.4.3. Mineria de dades (*data mining*)

La mineria de dades consisteix a extreure informació d'alt valor afegit (patrons ocults, tendències i correlacions) a partir de dades en brut. El que es pretén és descobrir informació estratègica per construir posteriorment models predictius sobre aquesta informació, models que provinguin dels productes de l'empresa, dels seus processos, dels seus clients, de la seva competència, etc., i així poder predir valors i tendències en el seu comportament. El concepte de mineria de dades no és nou, però l'aparició d'eines potents, fàcils d'utilitzar i a l'abast de tot tipus d'usuaris són elements recents que han contribuït a democratitzar i potenciar-ne l'ús.

Són autèntiques eines d'extracció de coneixement útil, a partir de la informació continguda en les bases de dades. El *data mining* incorpora la utilització de tecnologies basades en xarxes neuronals, arbres de decisió, regles d'inducció, anàlisi de sèries temporals i visualització de dades.

És fonamental entendre que *data mining* és un procés; no és simplement executar un determinat algoritme que fa alguna tasca, com, per exemple, una regressió lineal o una sèrie de càlculs i ja està. Això no és *data mining*, tot i que sí que es pot arribar a entendre com a anàlisi de dades.

El procés d'anàlisi de dades sol ser una tasca per a matemàtics i estadístics, però hi ha eines que faciliten aquest treball a usuaris de negocis o analistes. Apareix un nou perfil professional, denominat *data scientist*.

La mineria de dades és un concepte d'explotació de dades diferent als presentats anteriorment. Per començar, no es basa en coeficients de gestió o informació afegida, sinó que considera la informació detalladament continguda al magatzem de dades. De manera addicional, es persegueix identificar una relació o patró entre les dades, que tingui repercussions en el negoci més enllà d'una simple presentació d'informació.

Per exemple: com a enfocament empresarial, la mineria de dades permet comprendre múltiples problemes d'una organització: des de la segmentació de clients fins a l'optimització de la cadena de subministrament.

Actualment, els proveïdors de solucions de mineria de dades no s'estan aproximant al mercat amb una plataforma genèrica, sinó amb solucions específiques que resolen un problema d'una àrea i procés determinats, que s'integren en processos de negoci i que poden personalitzar-se a les necessitats d'una empresa concreta. Aquestes solucions busquen reduir el temps de desplegament del projecte, en reduir la personalització en un 50-60%. Aquest enfocament es coneix com a analítica de negoci. Cal comentar que, en l'àmbit dels negocis, no n'hi ha prou amb identificar patrons ocults en la informació, si després l'organització no pot actuar de manera àgil amb aquesta informació i accionar aquest coneixement. El fet rau en què aquelles organitzacions que són realment capaces d'executar una estratègia d'analítica de negoci correctament estan generant nous avantatges competitius.

9.4.4. Autoservei BI

Podem agrupar les diferents tendències actuals sota el paraigua que podríem denominar *self-service BI*. Trobem múltiples opcions: *data discovery, visual analytics, autodiscovery, query by exemple, discovery analysis, business discovery, natural analytics*, etc.

Les característiques més rellevants d'aquestes eines són que:

- Suporten tècniques d'**anàlisi visual** que faciliten la comprensió ràpida de les dades i permeten, així mateix, dur a terme presentacions clares i eficaces que ajudin en la presa de decisions.

- Faciliten l'anàlisi de **grans quantitats** de dades.

- Sovint, estan centrades en el *business discovery*: descobriment d'errors o oportunitats de millora.

- Utilitzen la **lògica associativa** (AQL), una tècnica que efectua les anàlisis i càlculs en memòria, obtenint amb això un temps de resposta excel·lent.

- Permeten fer **anàlisis interactives** recolzant-se en àgils funcionalitats de visualització i gestió de dades, la qual cosa fa possible una anàlisi lliure sobre el model de dades importat en l'eina. És a dir, l'usuari té la possibili-

tat d'interactuar amb les dades: comparar, filtrar, connectar unes variables amb altres, etc.

- Es recolzen en una **interfície intuïtiva** que facilita l'exploració de dades orientada tant a perfils TI com a analistes de negoci. L'eina s'adapta a l'usuari, no a l'inrevés.

- Agilitat i rapidesa en la manipulació de dades, recolzant-se en tecnologies ***in-memory***.

- S'adapten fàcilment a múltiples dispositius: ordinador, tauleta, telèfon intel·ligent, etc. (en anglès, ***responsive***).

Les principals diferències entre l'autoservei BI i el BI tradicional són les següents:

- Més flexibilitat en l'anàlisi, l'usuari decideix què utilitza com a Dimensió, indicador, jerarquies d'anàlisis, etc. Com que no existeix una capa semàntica, l'usuari s'enfronta directament a la BBD, la qual cosa pot ser ardu quan no hi ha gaire documentació i dona lloc a la creació de vistes resum (típic tauló) amb camps autoexplicatius partint de les taules origen.

- No cal crear repositoris de dades tipus *data warehouse*. Accés directe a fonts sense processos ETL previs. Això dona flexibilitat, però podem trobar-nos dades en brut poc validades.

- Més independència respecte a l'àrea de TI, però menys control sobre els entorns d'anàlisis. Allibera a TI de treballs recurrents de poc valor.

- Més adequat per a organitzacions petites, amb un volum i complexitat d'informació menors. Cicles curts de desenvolupament i costos més baixos (considerant costos de productes + consultoria, competeix amb soluciones BI *open source*).

- Cicles més curts en processos d'anàlisi *ad hoc*.

9.4.5. Sistemes de cerca empresarial en llenguatge natural

A mesura que les organitzacions generen més i més dades i informació, ha crescut la necessitat de capturar, extreure, consolidar i accedir a la informació de manera ràpida i eficient.

Una possible solució a l'accés de la informació la proporcionen els sistemes de cerca empresarial. Aquests sistemes tenen l'objectiu d'indexar i fer accessibles tots els continguts empresarials de múltiples fonts, incloent fitxers, bases de dades, intranets, sistemes de gestió documental, sistemes operacionals, EIS, *business intelligence*, correus electrònics i bases de dades.

És a dir, els usuaris poden preguntar-li a la plataforma i aquesta respon a les peticions amb gràfics: una barreja d'anàlisi i visualització de dades amb intel·ligència artificial. Aquest és un pas important dins del món del BI, en la carrera per facilitar l'ús de les dades.

Aquests sistemes han evolucionat no només per proporcionar accés a continguts específics mitjançant llenguatge natural, sinó també per correlacionar informació de diferents fonts d'informació i usar aquesta correlació en els resultats de les consultes.

Per exemple, la cerca del nom d'un client pot retornar la informació del perfil de client (del CRM), el contracte del servei que l'empresa té amb ell (del gestor documental) i la informació històrica de les seves compres (de l'ERP o la FIC).

9.4.6. *Big Data*

La visualització de dades massives es tractarà en una altra assignatura específica sobre aquest tema.

9.4.7. *Webhousing* i *mobile BI*

Des de fa uns anys, l'evolució de la tecnologia ha estat centrada en Internet. Amb l'arribada i la democratització de les tecnologies mòbils, qualsevol nova tecnologia busca estar present i vinculada tant a Internet com a la mobilitat.

- El concepte de *webhousing* neix de la confluència dels magatzems de dades amb Internet. Aquesta simbiosi crea un nou esquema d'informació en el qual els clients tenen a la seva disposició grans quantitats d'informació.

- Una de les tendències actuals és oferir el magatzem de dades com un servei (com, per exemple, Amazon Redshift), cas en el qual l'accés a les dades sempre seria per Internet. Igualment, existeix la possibilitat de no només oferir el *data warehouse* com un servei, sinó també la capa d'accés i anàlisi com a serveis, el que es coneix en anglès com a *BI as a service*. En aquests casos, les mateixes arquitectures descrites són vàlides, però és el proveïdor qui les suporta mitjançant *cloud computing*.

- El concepte de *mobile BI* fa referència a l'accés de les dades a través de dispositius mòbils, encapsulant aquests accessos en aplicacions específiques de valor afegit, com per exemple un comparador de preus.

Tot i que cada vegada l'usuari disposa de més flexibilitat, el desenvolupament de noves tècniques de visualització o representació i l'existència de més llocs web interactius poden causar problemes per a les organitzacions. Cada vegada és més fàcil per als usuaris, ja sigui per accident o a propòsit, distorsionar o tergiversar les dades i després fer que aquestes distorsions i interpretacions errònies estiguin àmpliament disponibles per a altres persones.

Per tant, cal que les organitzacions d'estadística tinguin una política clara sobre com aplicar i oferir noves tècniques de visualització o representació.

10. Eines de suport a la presentació de dades

Es poden mostrar les dades de moltes maneres diferents, des de senzills gràfics de barres a diagrames de dispersió més complexos, mapes temàtics i piràmides de població animades. No falten ajudes tècniques: llibres escrits entorn de la visualització de les dades; anotacions en pàgines web dedicades al tema; i una àmplia gamma de programari i programes descarregables, disponibles per a qualsevol finalitat.

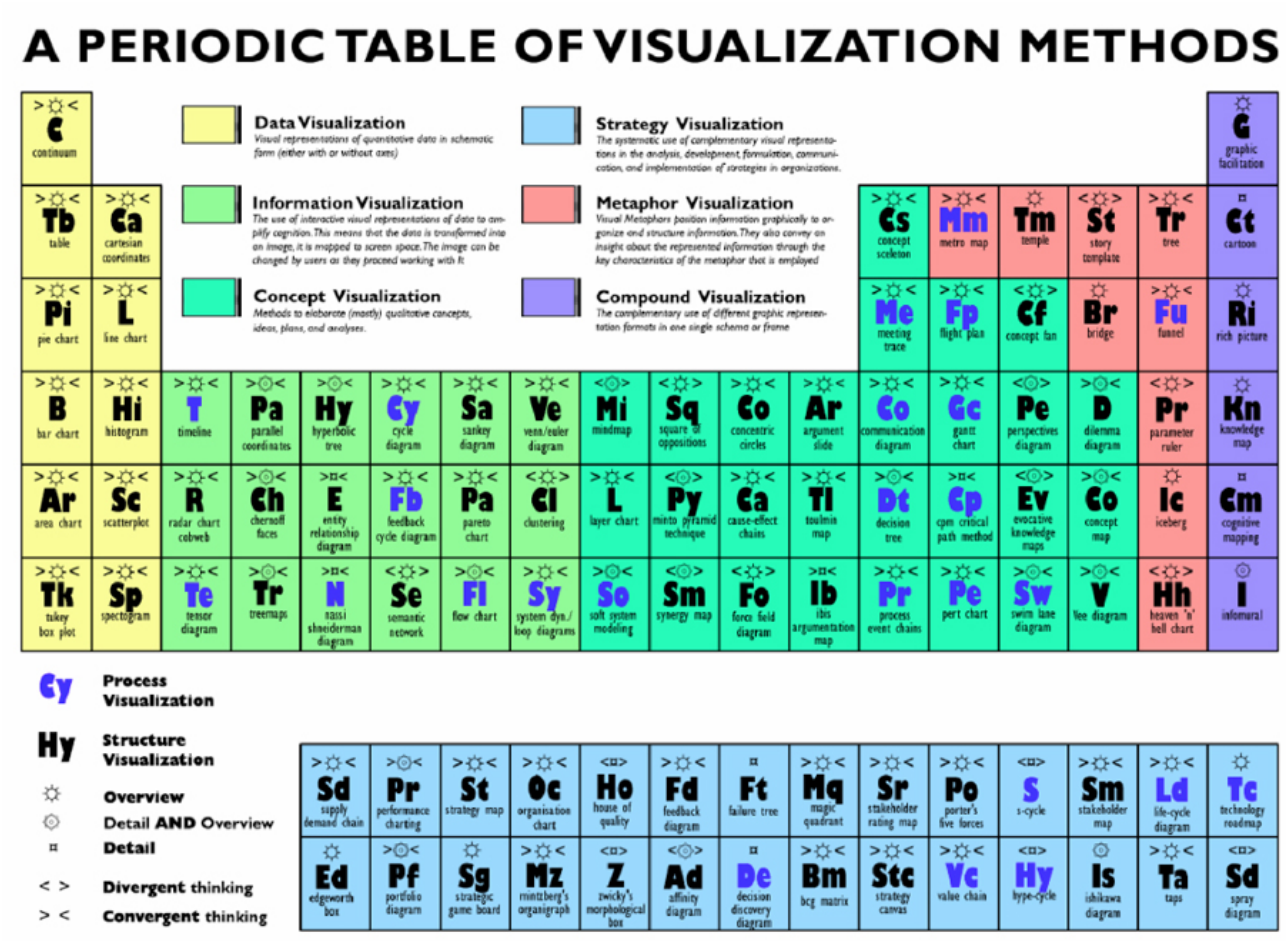
10.1. Metodologies

Són diferents els intents d'establir mètodes capaços de bregar amb els nous formats de dades i conjunts de dades: que permeten que formats de dades específiques, com espectrals o espaitemporals, puguin visualitzar-se adequadament; aconseguint adaptar la tecnologia per fer front a dades heterogènies (i de diferent credibilitat); i fent possible emfatitzar els aspectes que són rellevants per a la visualització.

1) Metodologia **Lengler i Eppler**:

Pretén definir i compilar mètodes de visualització existents, amb la finalitat de desenvolupar una revisió sistemàtica basada en la lògica i l'ús de la taula periòdica d'elements, revisada amb 100 mètodes de visualització i una proposta de com utilitzar cadascun d'aquests mètodes.

Figura 53. Taula periòdica de mètodes de visualització de dades

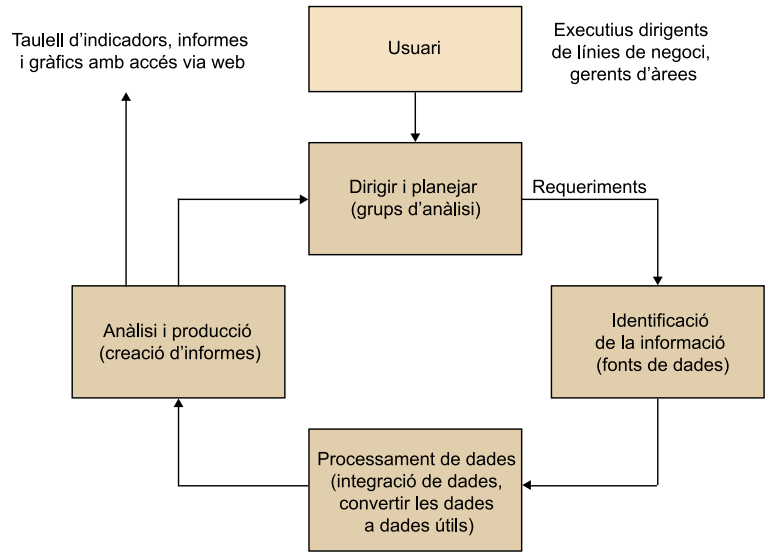


Fuente: http://www.visual-literacy.org/periodic_table/periodic_table.pdf

2) Metodologia **Lee Wittschen**:

Perquè BI doni suport a la presa de decisions, les dades han de ser les correctes i ser interpretades adequadament.

Figura 54. Procés Lee Wittschen



- Fase 1. Dirigir i planejar
  A través d'un grup d'analistes, es defineixen els requeriments per a la gestió de l'empresa i es formulen preguntes i hipòtesis per aconseguir l'objectiu i cobrir les necessitats.

- Fase 2. Recol·lecció d'informació
  Amb la finalitat d'obtenir els resultats esperats, s'han d'identificar les bases de dades on resideix la informació, sense importar que provingui de sistema externs.

- Fase 3. Processaments de dades
  S'han d'integrar les dades en un mateix format perquè puguin ser analitzades. Les dades de l'etapa 3 són interpretades i processades.

- Fase 4. Anàlisi i producció
  Es creen informes personalitzats, que traçaran els cursos d'acció que cal seguir, i proporcionen dades o indicadors rellevants per a la presa de decisions.

- Fase 5. Difusió
  Es generen els informes personalitzats i esperats, per a la seva interpretació, i es tracen i gestionen els cursos d'acció.

3) Metodologia **Ann K. Emery**:

La visualització de dades barreja tecnologia, ciència i creativitat a parts iguals. La personalització és el denominador comú, encara que no eximeix de l'observació d'algunes regles, com les propostes, que contribueixen a millorar els resultats. Per a una avaluació de la qualitat de la visualització de dades en major profunditat, es pot accedir a la *checklist* que proposen Ann K. Emery i Stephanie Evergreen, en la qual es tenen en consideració aspectes com:

La forma de presentació del text, en els casos en els quals sigui necessari incloure'l: haurà de ser llegible, concret i precís, preferiblement en disposició horitzontal i susceptible d'aplicar-se directament als elements que componen el gràfic o diagrama.

- La disposició gràfica: ha de ser clara, de proporcions adequades, amb un mínim de dues dimensions i sempre aplicant criteris d'ordre.

- L'ús del color: ha de tenir prou contrast, no perdre la seva capacitat d'informar, ni tan sols en el cas d'una impressió en blanc i negre, i ha de preveure les limitacions d'usuaris amb problemes de diferenciació de colors.

- Presència de línies: els llocs on estan presents i la seva aparença, que mai no ha de distreure l'atenció del contingut presentat.

- Visió global: que ha de permetre l'èmfasi en els detalls més rellevants, l'adequació del tipus de gràfic a les dades presentades, la contextualització i un nivell de precisió òptim.

10.2. Tècniques i components

Les eines i tècniques emergents estan proporcionant noves oportunitats per a la visualització de dades i per fer-les més interessants als usuaris.

Els generadors de taules dinàmiques, gràfics i mapes permeten als usuaris manipular les dades i crear les seves pròpies representacions. L'animació i el vídeo són formats atractius, semblants a la televisió. Fan una bona feina il·lustrant els canvis al llarg del temps, i inclouen descripcions verbals o textuals que expliquen el significat que hi ha darrere dels números. També estan sorgint nous tipus de visualitzacions, com *sparklines* i núvols d'etiquetes, que proporcionen alternatives per il·lustrar la informació.

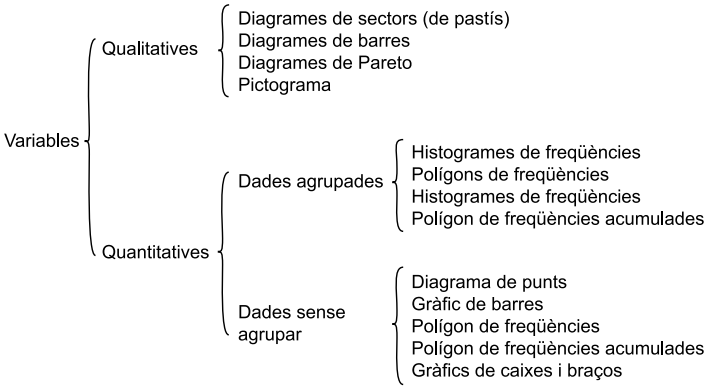
Sparklines

Són petits gràfics de línies de la mida de paraules, que mostren tendències en el temps.

No tots els tipus de gràfics són adequats per a un conjunt concret de dades. Alguns només valen per a un fi i d'altres s'adapten a diversos tipus de dades. Per prendre la decisió de quin utilitzar, hem de tenir en compte, d'una banda, el tipus de mesura utilitzada i, de l'altra, les característiques del conjunt de dades: si són sèries temporals o no, si intervé una o diferents variables...

Si el que desitgem és reflectir el **tipus de mesura** que estem utilitzant, triarem el tipus de gràfic segons aquesta figura:

Figura 55. Tipus de gràfic segons el tipus de mesura

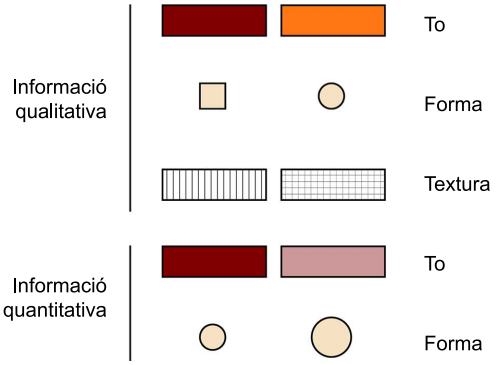


També podem utilitzar aquesta «recepta ràpida» d'aplicació general (encara que sempre amb excepcions, ja que la representació i simbolització conté, no ho oblidem, elements subjectius), segons aquests punts:

- Per a les **variables qualitatives**, s'utilitzen les variables visuals de color, forma i textura, en la mesura en què sigui possible, segons el tipus d'objecte geomètric que cal simbolitzar.

- Per a les **variables quantitatives**, el valor del color i la mida són les més adequades, i aquesta última és l'única que permet transmetre tota la informació en el cas de variables de tipus raons. El to de color pot emprar-se, però s'ha de triar una gamma de tons que presenti algun tipus de lògica que permeti establir un ordre.

Figura 56. Forma, color i textura segons el tipus de variable



L'**animació** i el vídeo són dues importants tècniques emergents de visualització de dades. Quan es té en compte la popularitat de la televisió i el cinema, no és d'estranyar que als usuaris els agradi la idea de rebre missatges a través d'imatges en moviment. Aquest format fa més fàcil explicar la història, mitjançant la combinació de descripcions en àudio o text, amb il·lustracions gràfiques que expliquen el significat que s'amaga darrere dels números.

Un bon exemple de l'ús d'animacions per explicar estadístiques és la piràmide de població del Regne Unit, 1971-2081

https://www.ons.gov.uk/peoplepopulationandcommunity/populationandmigration/populationestimates/articles/ukpopulationpyramidinteractive/2020-01-08

Les piràmides dinàmiques de població, desenvolupades per diverses organitzacions d'estadística, incloses l'Oficina d'Estadístiques Nacionals del Regne Unit i Estadístiques del Canadà, són bons exemples de la combinació d'animació amb interactivitat en una interfície senzilla. Els usuaris poden fer clic per veure com la forma de la piràmide de població canvia amb el pas del temps. També poden interactuar amb el gràfic seleccionant grups d'edat i fixant-se detalladament en els números i en les proporcions en la població total.

Hans Rosling, cofundador de Gapminder, ha tingut un gran èxit amb l'ús de l'animació per il·lustrar dades. Rosling ha aconseguit una audiència massiva a través de petits vídeos en línia (*gapcasts*), que són conferències sobre diferents temes, com la mortalitat materna, la globalització, l'energia, etc. (http://www.gapminder.org/video/gap-cast/).

Les ***sparklines*** tenen l'avantatge de mostrar una gran quantitat d'informació d'un cop d'ull, i es poden col·locar al costat de paraules que expliquin el seu significat. Tufte (2006) va ser el primer a proposar-les. Aquests «directes i senzills gràfics de la mida de paraules» milloren la presentació de les dades amb una representació visual, sense ocupar gaire d'espai».

La següent figura mostra un exemple de *sparklines* utilitzades per il·lustrar les fluctuacions de les tasques efectuades pels membres d'un equip de projectes.

Figura 57. *Sparklines* elaborades en MS Excel

Team Member	Total Tasks Completed	w1	w2	w3
Julie	46%	13	15	19
John	45%	11	18	11
Jabba the hut	-20%	15	14	14
Johnson	6%	18	17	14
Jeremy	43%	14	20	10
Josh	-33%	15	12	19

Un **núvol d'etiquetes** (a vegades també denominat núvol de paraules) és una representació visual de la freqüència de paraules o etiquetes en un text o *dataset* concret. Es pot veure, sovint, en pàgines web com una llista de categories, on cada paraula és un vincle que condueix a l'usuari a més informació relacionada amb aquesta paraula. Els núvols d'etiquetes són una forma útil d'identificar els termes comuns d'un text i construir taxonomies de paraules clau.

Exemple

L'exemple següent ha estat creat utilitzant text d'alguns apartats d'aquest mòdul i il·lustra clarament les paraules clau.

Figura 58. Núvol de paraules

10.3. Recursos en línia

Ja sabem que la imatge és un element molt poderós. El 90% de la informació que es transmet al cervell és visual, i aquesta es processa 60.000 vegades més ràpid que el text, per la qual cosa és lògic que acompanyem les nostres dades amb imatges o recursos gràfics adequats.

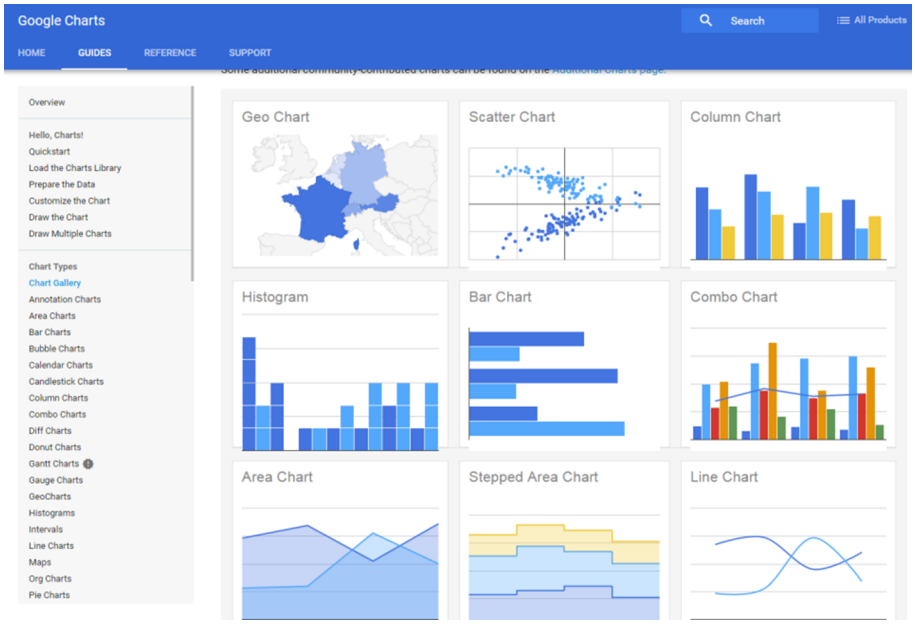
Existeixen multitud de recursos disponibles:

a) Guies de **cromatisme**, **estils** i **tipus de gràfics**:

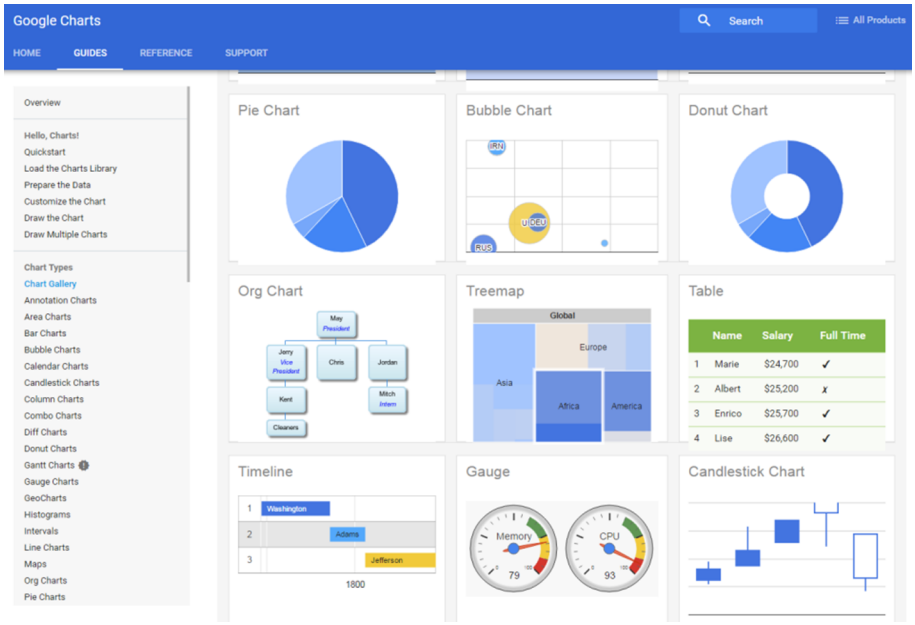
- *Color advice for cartography* (http://colorbrewer2.org).

- Tipus de gràfics, quin utilitzo? (https://www.ine.es/explica/explica_pasos.htm).

- *Google Chart types*

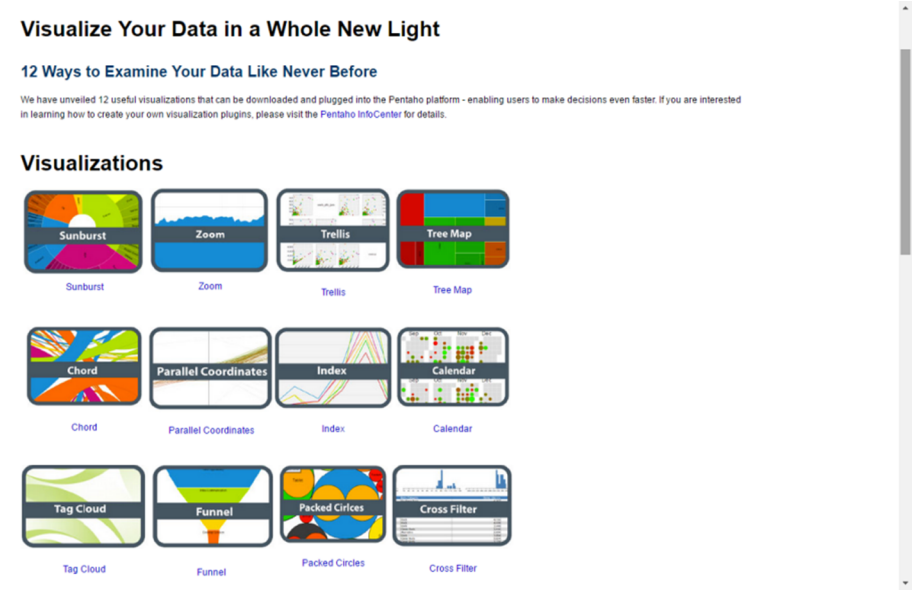


Font: https://developers.google.com/chart/



Font: https://developers.google.com/chart/

- *Pentaho Visualization plugins examples*



Font: https://pentaho-public.atlassian.net/wiki/spaces/COM/pages/205433892/Visualization+Plugins

b) Bancs d'**imatges** i **recursos gràfics**:

Recopilem, a continuació, diversos llocs web en els quals es poden trobar recursos gràfics per fer infografies o il·lustracions.

- Free Digital Photos (http://www.freedigitalphotos.net/).
- FreeImages (http://www.freeimages.com/).
- Morguefile (http://www.morguefile.com/).

- Icojam (http://www.icojam.com/).
- Freepik (http://www.freepik.es/).
- Flaticon (http://www.flaticon.com/).

c) Bancs de **tipografies**:

- Dafont (http://www.dafont.com/es/).
- Google Fonts (https://www.google.com/fonts):

d) Llibreries gràfiques i de components:

- *D3: collection of simple charts made with d3.js*. (https://d3-graph-gallery.com/).
- *Mondrian: general purpose statistical data-visualization* (http://www.theusrus.de/Mondrian/).

e) Comunitats entorn de les dades:

Existeixen llocs web que estan afegint una nova dimensió a les presentacions visuals, a través de la construcció de comunitats en línia sobre la visualització i l'intercanvi de dades. Aquests llocs permeten als usuaris carregar conjunts de dades i crear gràfics, per al seu intercanvi i discussió amb altres usuaris. Altres aplicacions, denominades aplicacions web híbrides (*mashups*), combinen dades o funcionalitats de dues o més fonts per crear un servei nou.

> Per exemple, des del laboratori d'investigació d'experiència d'usuari col·laborativa d'IBM (CUE - Collaborative User Experience) posen també el focus en la visualització de dades, no només sobre gràfics de barres i gràfics circulars, sinó com d'efectiva pot ser la informació visual en gràfics de múltiples variables. Es disposa d'un programari que està disponible en estil Web2.0 sense càrrec, simplement carregant un conjunt de dades i escollint un dels 16 estils de presentació diferents. https://www.ibm.com/support/pages/many-eyes-and-visualization-data

Encara que fins ara l'èxit ha estat desigual, aquest tipus de comunitats en línia, sens dubte, proporcionen una forma relativament fàcil d'arribar a més usuaris i és, per tant, una àrea emergent en la visualització de dades que cal tenir en compte.

10.4. Eines de visualització

En l'actualitat, hi ha una gran varietat d'eines analítiques i de BI, la qual cosa pot dificultar l'elecció de la solució més adequada, que en qualsevol cas tindrà a veure amb el tipus d'empresa, mida, sector, objectius i processos que es vulguin cobrir.

Els «quadrants màgics» són una eina creada i promoguda per l'empresa Gartner, consistent en una representació gràfica del mercat, analitzat en un moment determinat.

D'acord amb el definit per Gartner, els líders són aquells fabricants que tenen una visió clara de la direcció del mercat i que desenvolupen competències per mantenir la seva posició actual.

Aquests quadrants ajuden les organitzacions a identificar i poder diferenciar els proveïdors de serveis en el camp BI i plataformes analítiques. Han de ser considerats com una eina, i no com una guia.

No és fàcil trobar informació clara sobre tarifes de llicenciament i suport de programari comercial de *business intelligence*. En aquest enllaç, trobem un estudi que ens pot orientar:

http://www.slideshare.net/oktopuslu/bi-comparison-of-open-source-and-traditional-vendor-4327259

No sempre s'han considerat, en les comparatives d'eines, les opcions de BI de codi obert (*open source*) que existeixen al mercat, enfront de les opcions de BI de propietari.

- Les primeres, sense cost d'adquisició, però amb una quantitat important d'hores de desenvolupament.

- I les segones, amb el valor afegit d'un suport tècnic molt interessant, que en les *open source* s'ofereix a través de les comunitats d'usuaris que han adoptat aquestes eines.

Tenint en compte tot això, triar entre eines de visualització de dades redueix la seva complicació, ja que la meta és clara: ja se sap el que es necessita.

És el moment d'avaluar altres variables, en funció dels criteris següents:

1) Criteris de negoci:

- Velocitat d'implementació.

- Escalabilitat.

- Preu.

- Cost de llicències/suport tècnic.

- Viabilitat a llarg termini.

2) Criteris tècnics:

- Orientació a usuaris finals.

- Capacitats analítiques.

- Visualització interactiva.

- Capacitats de modelatge.

- *Drill-down* visual.

- Controls visuals.

- Suport d'escriptori.

- Entorn de desenvolupament.

- Integració amb altres eines/plataformes/sistemes.

- Conjunts de dades *in-memory*.

En aquest exemple, veiem alguns criteris i factors que podem tenir en consideració quan avaluem diverses tecnologies, abans de la seva elecció:

- Ha tingut un creixement molt espectacular en els últims anys?
- Utilitza lògica associativa (AQL), fent les anàlisis i càlculs en memòria?
- Permet l'elaboració de prototips ràpids?
- Suporta grans volums de dades (*big data*)?
- Forma part d'un paquet ofimàtic de BI complet o és una solució específica?
- La relació preu-suport és interessant?
- En els últims anys, ha anat escalant posicions en els quadrants de Gartner?
- Cobreix les funcionalitats requerides?
- Existeixen altres organitzacions que utilitzen aquesta eina?
- Com gestiona les metadades?
- Disposa d'un bon sistema de versionament i control del codi font?
- Quina política de noves versions, millores i resolucions de *bugs* aplica?
- És accessible i disposa de visualització des d'iPad, Android o altres?
- Disposa d'un sistema que permeti fer *write-back* per establir pressupostació, *forecasts*, simulació o regles de negoci?
- Disposa d'un motor OLAP?
- Disposa d'eines de *data mining*?
- Suporta diferents arquitectures i entorns com Windows o Linux?
- Pot ser implementada i utilitzada durant setmanes, depenent de la complexitat i el volum de dades?

10.5. Eines de suport

Trobem diferents eines que complementen les funcionalitats de les eines de l'apartat anterior i que cobreixen tasques tècniques d'optimització o refinament per al seu bon funcionament.

- **Optimitzador de consultes (*query analyzer*)**
  És una eina d'optimització de consultes. Amb ella es poden visualitzar i solucionar problemes dels registres de consulta generats a l'hora d'executar informes de consulta dinàmica. Algunes bases de dades incorporen un planificador de consultes que duu a terme funcions equivalents.

- **Agregador de dades**
  L'assessor d'agregació analitza cubs dinàmics per recomanar agregats que millorin el rendiment del cub. L'assessor d'agregació pot aplicar els agregats en bases de dades automàticament a les taules de bases de dades i al model. Pot decidir si desitja aplicar els agregats a bases de dades immediatament, o planificar-los per a més endavant. Quan aplica recomanacions en bases de dades, l'assessor d'agregació pot planificar un esdeveniment per carregar dades més tard.

- **Integració/connectivitat**
  Ineludiblement hem de manipular múltiples fonts i formats de dades a les bases de dades. La majoria de les eines accepten formats com Excel, Access i text; poden accedir a moltes bases de dades com Microsoft SQL Server, MySQL, Oracle o Greenplum; i també tenen la possibilitat d'utilitzar les API (*application programming interface*) d'altres eines, per a l'extracció sistemàtica de dades; conjunts de dades d'Hadoop, dades de *streaming* i serveis al núvol.

- **Portals de distribució, col·laboració i alertes**
  Aquests sistemes envien alertes i correus electrònics de manera automàtica per notificar els esdeveniments als responsables de la presa de decisions de la seva organització, a mesura que tenen lloc, perquè puguin prendre les decisions de manera ràpida i eficaç. També es poden veure i obrir espais de treball on compartir i manipular dades o anàlisis. Hi podrem utilitzar comentaris, activitats i programari social per a la presa de decisions col·laborativa.

Resum

A vegades, per desenvolupar un determinat tipus d'aplicacions, hem d'aplicar tècniques específiques de disseny. Aquest és el cas de les eines OLAP, utilitzades per fer l'anàlisi multidimensional. El seu disseny es basa en la definició de Fets objecte d'anàlisi i les Dimensions utilitzades per a analitzar-los.

En aquest mòdul, hem vist quins són els elements principals d'un model multidimensional: quines estructures de dades ofereix, quines operacions podem fer amb les dades i quines restriccions d'integritat podem definir. També hem estudiat com es fa un disseny conceptual multidimensional pas a pas i com es pot implementar en un SGBD relacional. Heu conegut algunes tècniques d'accés per millorar el temps de resposta i les noves paraules reservades definides a la SQL'99 per facilitar l'escriptura de les consultes.

Finalment, es presenta de manera pràctica la part «visible» i més important del DW: l'explotació de les dades. Es revisen les diferents formes de presentació (informes clàssics, cubs OLAP, quadres de comandament i altres formes de visualització), bones pràctiques o eines de presentació.

Activitats

1. Llegiu les divuit regles d'avaluació d'eines multidimensionals del Dr. E. F. Codd. Les podeu trobar al llibre d'E. Thomsen *OLAP Solutions*.

2. Podeu veure com un determinat SGBD (per exemple, SQL Server de Microsoft) implementa les operacions ROLLUP i CUBE de l'estàndard SQL'99.

Exercicis d'autoavaluació

1. Feu el disseny multidimensional per al cas d'un conjunt de magatzems de productes d'un distribuïdor de supermercats. Aquest distribuïdor serveix com a intermediari entre els proveïdors i les botigues. Estem especialment interessats a analitzar el següent:

a) La quantitat de cada producte (estoc) que tenim quan acaba el dia a cadascun dels magatzems, tenint en compte que podem haver comprat el mateix producte a diferents proveïdors.

b) Cada moviment que hi ha als magatzems (independentment de si és una entrada –què ens serveix un proveïdor– o una sortida –cap a una botiga–), segons el producte, l'hora en què té lloc i l'entitat (ja sigui proveïdor o botiga) que el genera.

Per a cada moviment tenim un albarà amb un codi que l'identifica de manera vaig unívoca, assignat pel sistema operacional de gestió dels magatzems comú a tota l'empresa. Un albarà conté les dades del magatzem corresponent, l'hora de lliurament i el nom del proveïdor o botiga, juntament amb la llista de productes que se serveixen.

Cada magatzem (de la mateixa manera que els proveïdors i les botigues) està en una població i cada població pertany a una regió. Tant magatzems com botigues i proveïdors s'identifiquen per un nom. Per als magatzems, també disposem dels metres quadrats de superfície.

De la Dimensió temporal, ens interessen els mesos, els trimestres, els quadrimestres i els anys. A més d'una hora, dia, mes, trimestre, quadrimestre o any qualsevol, també volem poder seleccionar els dies festius i els mesos d'estiu.

Els productes estan identificats pel codi de barres, però també volem veure la seva descripció. Interessa agrupar els productes en tipus de producte, que s'identifiquen pel seu nom. Pensem que un producte no pot ser de més d'un tipus.

2. Estimeu l'espai que ocupa l'esquema anterior. Penseu que estem interessats en les dades corresponents als últims tres anys (mil dies aproximadament), el nostre catàleg conté 10.000 productes diferents (dividits en 1.000 tipus), tenim deu magatzems, servim a 1.000 botigues i comprem a 1.000 fabricants. De mitjana, a cada fabricant li fem una comanda setmanal, mentre que cada botiga ens fa una comanda dia sí, dia no. En total, en acabar l'any tenim 235.000 albarans, amb una mitjana de 50 productes per albarà.

3. Tenint en compte l'esquema i els volums de dades anteriors, digueu quin tipus d'eina OLAP seria millor utilitzar (ROLAP, MOLAP, etc.).

4. Traduïu l'esquema anterior a relacional.

5. Penseu ara que per millorar el temps de resposta de les consultes en la relació `EstocDiari(Producte,Dia,Magatzem,quantitat,estoc)` que conté l'estoc dels 10.000 productes (de 1.000 tipus), els 1.000 dies, als 10 magatzems (repartits en 3 regions), també decidim emmagatzemar les tres relacions següents, que contenen dades preagregades de les Cel·les corresponents:

Estoc1(Producte,Dia,Regio,quantitat,estoc)
Estoc2(Producte,Any,Magatzem,quantitat,estoc)
Estoc3(Tipus,Any,Magatzem,quantitat,estoc)

a) A quantes tuples cal accedir per resoldre una consulta que demana l'estoc anual d'un determinat producte en una regió, si el calculem accedint a `Estoc`? A quantes accedint a `Estoc1`? A quantes accedint a `Estoc2`? A quantes accedint a `Estoc3`?

b) Quina de les quatre relacions hauríem d'utilitzar per saber el total d'articles (de tots els productes) que hem tingut en estoc durant tots els anys a totes les regions?

c) Quines Cel·les podem calcular a partir de `Estoc3`?

6. Trobeu una seqüència d'operacions multidimensionals que us permetin passar del primer cub al segon. Tots dos pertanyen a l'Estrella `Inventari` dissenyada anteriorment.

Origen				
Quantitats de cada tipus de producte per albarà	1.111	2.222	3.333	4.444
Materials d'oficina	1	2	3	4
Aliments	4	3	2	1

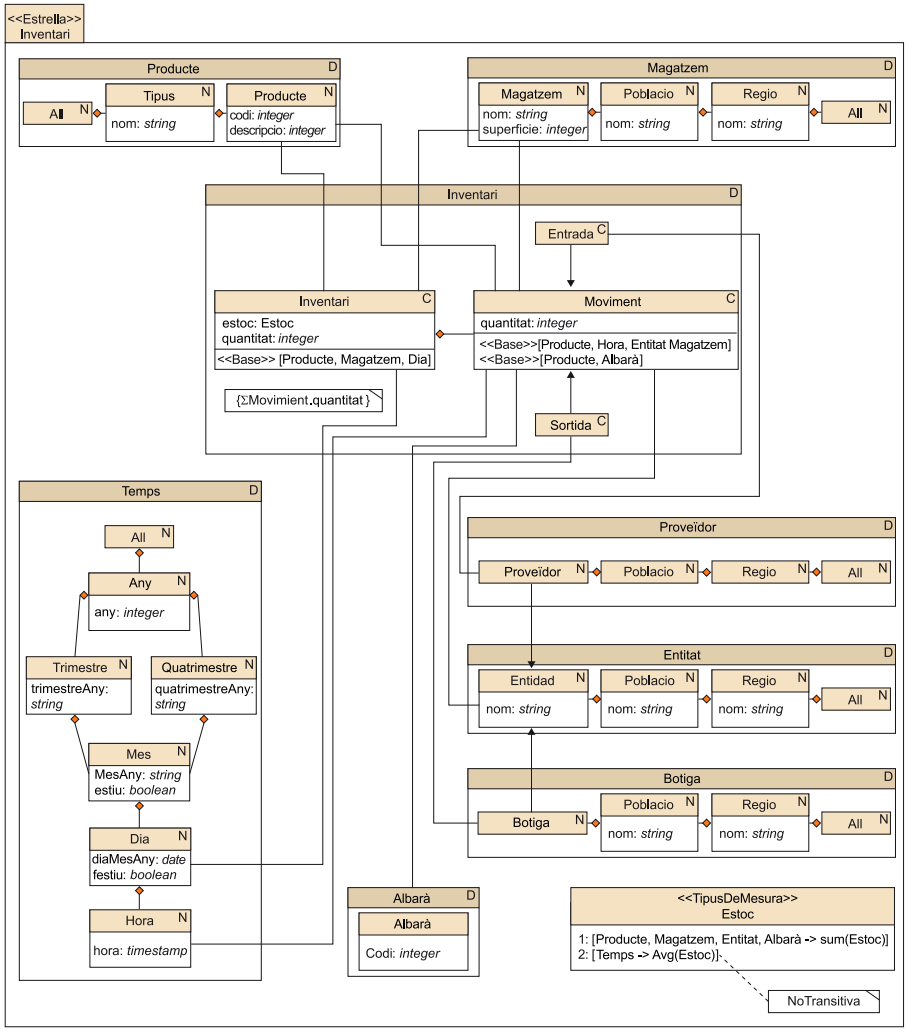
Destí					
Quantitats de producte per regió de magatzem i dia		23-4-2002	24-4-2002	25-4-2002	26-4-2002
Catalunya	**Gomes**	1	0	0	2
	Bolígrafs	0	2	3	2

7. Feu una consulta de la taula Entrada definida a l'exercici 4, utilitzant les paraules reservades definides a l'estàndard de 1999, per obtenir el resultat següent escrivint el mínim possible:

Regió de magatzem	Any	Tipus de producte	Quantitat
Catalunya	2001	Materials d'oficina	100
Catalunya	2001	Aliments	200
Catalunya	2001	Null	300
Catalunya	2002	Materials d'oficina	400
Catalunya	2002	Aliments	500
Catalunya	2002	Null	900
Catalunya	Null	Materials d'oficina	500
Catalunya	Null	Aliments	700
Catalunya	Null	Null	1.200

Solucionari

1. La quantitat que hi ha al magatzem pot provenir de diferents fabricadors. Per tant, no podem associar la Dimensió `Fabricant` amb la Cel·la `EstocMensual`.



2. Una fita màxima de l'espai de la Cel·la `Moviment` és 4.800.000.000.000 cel·les (10.000 productes × 24.000 hores × 2.000 entitats × 10). Ara bé, podem obtenir una fita més petita si la calculem a partir del nombre d'albarans. D'aquesta manera, obtenim que l'espai té com a màxim 7.050.000.000 cel·les (3 anys × 235.000 albarans/any × 10.000 productes). No obstant això, com que l'enunciat diu que hi ha una mitjana de 50 productes a cada albarà, sabem que realment seran només 35.250.000 cel·les. Si pensem que cada quantitat ocupa 2 *bytes* i que necessitarem cinc identificadors (un per a cada Dimensió) de 4 *bytes* cadascun, la Cel·la ocuparia 775.500.000 *bytes* (775,5 Mb).

La Cel·la `EstocDiari` defineix un espai màxim de 100.000.000 cel·les (10.000 productes × 10 magatzems × 1.000 dies). Pensem que cada estoc ocupa 2 *bytes* i que necessitarem tres identificadors de 4 *bytes* cada un. La Cel·la ocuparia 1.600.000.000 *bytes* (1.600 Mb).

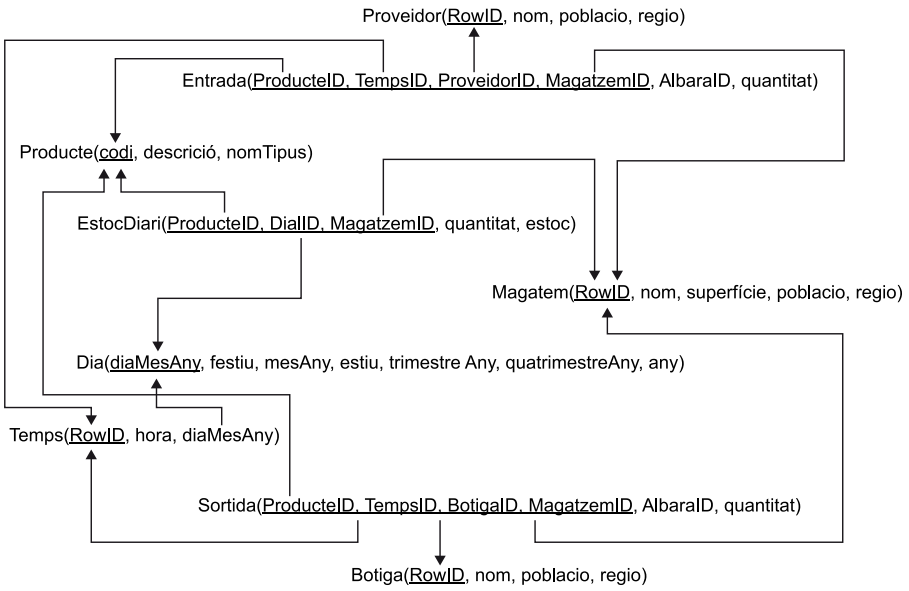
En total, tota l'Estrella ocuparà 2375,5 Mb (si negligim l'espai que puguin ocupar les Dimensions).

3. Les dades corresponents als moviments són molt disperses. A cada hora no ens arriben comandes de tots els clients ni ens serveixen tots els proveïdors. Probablement, la millor elecció seria una eina ROLAP, pel gran volum de dades que suporten i perquè no tenen problemes amb la dispersió. Per contra, la dispersió del cub corresponent a `EstocDiari` és pràcticament 1. Totes les cel·les que pot haver-hi gairebé sempre estaran presents. Una eina MOLAP que suporti aquest volum de dades seria la millor opció, perquè probablement ens oferirà un

temps de resposta menor que una implementació ROLAP. Si volem tenir els dos cubs dins del mateix sistema, hauríem de triar un HOLAP.

4. `Albara` és una Dimensió degenerada, que quedarà simplement com a part d'una clau alternativa. No desapareix del tot, perquè ens pot interessar fer consultes per albarà. A més d'això, com que les subclasses de Moviment no tenen cap atribut específic, pel que fa a l'espai que ocuparà, és equivalent tenir una superclasse o dues subclasses. Per tant, ens hem de fixar simplement en el temps de resposta.

Serà més ràpid consultar una taula gran o dues de petites? En aquest cas, implementarem l'especialització com si fos una partició horitzontal. Cada tipus de moviment està en una taula diferent (si els volem tots, només cal unir-los).



5.

a) En el pitjor cas, serà necessari accedir a totes les tuples de cada relació. `EstocDiari` conté 10^8 tuples, `Estoc1` en conté 3×10^7 i `Estoc2` en conté 3×10^5. `Estoc3` conté només 3×10 tuples, però no ho podem utilitzar perquè les seves dades no són prou detallades (estan en granularitat tipus de productes i volem consultar productes).

b) Tot i que seria molt més eficient utilitzar `Estoc2` o `Estoc3`, hauríem d'utilitzar `Estoc1`si volem tenir el resultat exacte. Recordeu que al llarg del temps agreguem utilitzant la mitjana i que la mitjana no és una funció transitiva. D'aquesta manera, hem d'accedir a les dades en el nivell més detallat, que en aquest cas és `Dia`.

c) Considerant que no podem continuar agregant al llarg de la Dimensió `Temps` per la raó esmentada anteriorment, i que els tipus de producte són disjunts (com diu l'enunciat), podríem calcular les Cel·les següents:

Estoc4(Tipus,Any,Poblacio,quantitat)
Estoc5(Tipus,Any,Regio,quantitat)
Estoc6(Tipus,Any,quantitat)
Estoc7(Any,Magatzem,quantitat)
Estoc8(Any,Poblacio,quantitat)
Estoc9(Any,Regio,quantitat)
Estoc10(Any,quantitat)

6. Una possible solució podria ser la següent:

A := $Selecció_{\text{Tipus.nom="Materials d'oficina"}}(origen)$
B := $CanviBase_{\text{Producte×Magatzem×Temp×Entitat}}(A)$
C := $Drill\text{-}down_{\text{Producte::Producte}}(B)$
D := $Roll\text{-}up_{\text{Entitat:All}}(C)$
E := $Roll\text{-}up_{\text{Magatzem::Regió}}(D)$
F := $Roll\text{-}up_{\text{Temps::Dia}}(E)$
$desti$:= $CanviBase_{\text{Magatzem×Producte×Temps}}(F)$

7.

```
SELECT m.regio, d.any, p.tipus, SUM(f.quantitat)
FROM entrada f, magatzem m, temps t, dia d, producte p
WHERE f.MagatzemID=m.RowId
AND f.TempsID=t.RowID
AND t.diaMesAny=d.diaMesAny
AND f.ProducteID=p.codi
AND d.any IN (2001,2002)
AND p.tipus IN (“Aliments”, “Materials d’oficina”)
AND m.regio=”Catalunya”
GROUP BY m.regio, CUBE(d.any, p.tipus)
ORDER BY m.regio, d.any, p.tipus;
```

Glossari

dada *f* Mesurament. Observació feta i emmagatzemada en algun sistema.

descriptor *m* Atribut d'una Dimensió utilitzat per seleccionar instàncies dels Fets.

dice *f* Operació multidimensional que redueix la mida de l'espai que selecciona valors en les Dimensions.

dimensió *f* Punt de vista utilitzat en l'anàlisi d'un Fet determinat.

dispersió *f* Quocient entre el nombre màxim teòric d'instàncies que pot arribar a haver-hi i les instàncies que hi ha realment en un cub.

drill-across *m* Operació multidimensional que, tenint en compte un espai, canvia les dades que es mostren. Canvia d'un tema d'anàlisi a un altre.

drill-down *m* Operació multidimensional que mostra les dades més detallades.

entitat-interrelació
sigla **ER**
en entity-relationship

factora d'informació corporativa *f* Conjunt d'elements de programari i maquinari que ajuden en l'anàlisi de dades per prendre decisions.
sigla **FIC**

fet *m* Objecte d'anàlisi.

finestra d'actualització *f* Període de temps dedicat a l'actualització d'un magatzem de dades.

granularitat *f* Mida d'un objecte respecte a un altre.

HOLAP *m* Eina OLAP que barreja característiques ROLAP i MOLAP.
en hybrid OLAP

informació *f* Dades rellevants per a algun decisor que afecten alguna de les seves decisions.

jerarquia d'agregació *f* Conjunt de relacions entre les instàncies d'una Dimensió que indica com n'agrupem algunes per obtenir-ne d'altres.

magatzem de dades corporatiu *m* Conjunt de dades integrades i històriques de tota l'empresa.

magatzem de dades departamental *m* Conjunt de dades que resol les necessitats d'anàlisis d'un cert departament o conjunt d'usuaris.

mesura *f* Dada numèrica associada a un esdeveniment que volem analitzar.

metadada *f* Dada sobre les dades.

MOLAP *m* Eina OLAP que utilitza matriuss *n*-dimensionals (en lloc de taules relacionals) per emmagatzemar les dades.
en multidimensional OLAP

nivell d'agregació *m* Conjunt d'instàncies de la mateixa granularitat dins d'una Dimensió.

O ^{3}LAP *m* Eina OLAP implementada sobre un SGBD orientat a l'objecte.
en object-oriented OLAP

OLAP *m* Sigles que fan referència a les eines d'anàlisi, normalment multidimensional. Categoria de tecnologia de programari que permet als analistes, gestors i executius millorar el seu coneixement de les dades mitjançant l'accés ràpid, consistent i interactiu a una àmplia varietat de possibles vistes d'informació que ha estat transformada des de les dades operacionals per reflectir la dimensionalitat real de l'empresa com l'entén l'usuari (The OLAP Council).
en on-line analytical processing

OLTP *m* Sigles que fan referència a sistemes operacionals, que ajuden en el dia a dia de la nostra empresa.

en on-line transaccional processing

ROLAP *m* Eina OLAP implementada sobre un SGBD relacional.

roll-up *m* Operació multidimensional que resumeix les dades augmentant la granularitat al llarg d'una Dimensió.

SGBD *m* Vegeu **sistema de gestió de bases de dades**.

sistema de gestió de bases de dades *m* Programari que gestiona i controla bases de dades. Les seves funcions principals són les de facilitar l'ús simultani a molts usuaris de diferents tipus, independitzar a l'usuari del món físic i mantenir la integritat de les dades.
sigla **SGBD**
en database management system

sistema operacional *m* Sistema que ajuda en les operacions diàries de negoci d'una organització.

sistema transaccional *m* Sistema basat en transaccions de lectura/escriptura.

slice *f* Operació multidimensional que redueix la dimensionalitat de l'espai fixant un valor en una Dimensió.

VOLAP *m* Eina OLAP que emmagatzema les dades per columnes en lloc de per files.
en vertical OLAP

Bibliografia

Cano, J. L. (2007). *Business intelligence. Competir con información*. Dipòsit legal: M-41185-2007.

Emery, A. K. (2016). *Data visualization checklist*. http://annkemery.com/portfolio/data-viz-checklist/

Kimball, R. (2010). *The Kimball Group Reader; Relentlessly Practical Tools for deta warehousing and Business Intelligence*. Indianapolis: John Wiley & Sons, Inc.

Lantares Solutions.*Visualización de Datos*.

Lengler, R.; Eppler, M. J. (2007). *Towards a Periodic Table of Visualization Methods for Management*. Lugano: Institute of Corporate Communication, University of Lugano.

Mendack, S. (2008). *OLAP without cubes: Data Analysis in non-cube Systems*. Hoboken: VDM Verlag.

Naciones Unidas (2009). *Cómo hacer comprensibles los datos: Una guía para presentar estadísticas*. Ginebra.

Third Nature (2010). *Lowering the Cost of Business Intelligence with Open Source*. Technology White Paper.

Webb, C. i altres (2006). *MDX Solutions: With Microsoft SQL Server Analysis Services 2005 and Hyperion Essbase*. Hoboken: John Wiley & Sons.

Wrembel, R. (2006). *Datawarehouses and OLAP: Concepts, Architectures and Solutions*. Hersbey: IGI Globals.

http://www.gestiopolis.com/inteligencia-de-negocios/

http://www.kimballgroup.com