

DISEINU FISIkoa

5. gaia: Diseinu fisikoa

- *Helburua*
- *Datu-baseen gordetzea eta fitxategi-antolamendua*
Elmasri & Navathe 09
- *Fitxategietako indexatze-egiturak edo atzipen-egiturak*
Elmasri & Navathe 09
- *Diseinu fisikorako gomendioak eta diseinua nola egin*
Conolly & Begg 05
- *Kontsulta-optimizazio*
Elmasri & Navathe 09

5. gaia: Bibliografia

- Elmasri, R. & Navathe, S.B.

Datu-base sistemak. Oinarriak.

Euskal Herriko Unibertsitateko Argitalpen Zerbitzua (2009).

- Jatorrizko bertsioa – ingelesez; gazteleraz ere badago.

- Connolly, T.M. & Begg, C.E.

Sistemas de bases de datos. Un enfoque práctico para diseño, implementación y gestión.

Pearson. Addison Wesley (2005).

- Jatorrizko bertsioa – ingelesez.

Diseinu fisikoaren **helburua**

- Aplikazio ezberdinetan errendimendu ona lortzeko helburuarekin, DBa osatzen duten fitxategientzako **biltegiiratze-egitura (fitxategien antolamendua)** eta **atzipen-egitura (bilaketa modua)** egokienak aukeratzeko dira.

Diseinu fisikoa. Erabili beharreko irizpideak

- Erantzun-denbora
 - Transakzioa bidaltzen denetik erantzuna jaso arte pasatzen den denbora
- Erabilitako espazioa
- Transakzioen "Throughput" (Produktibitatea)

Diseinurako erabakiak hartzeko normalean tarteko kasua eta kasurik okerrena hartzen dira kontuan.

Simulazio-teknikak erabiltzen dira.

Datu-baseen gordetzea. Biltegitratze motak

- Lehen mailako gordetzea

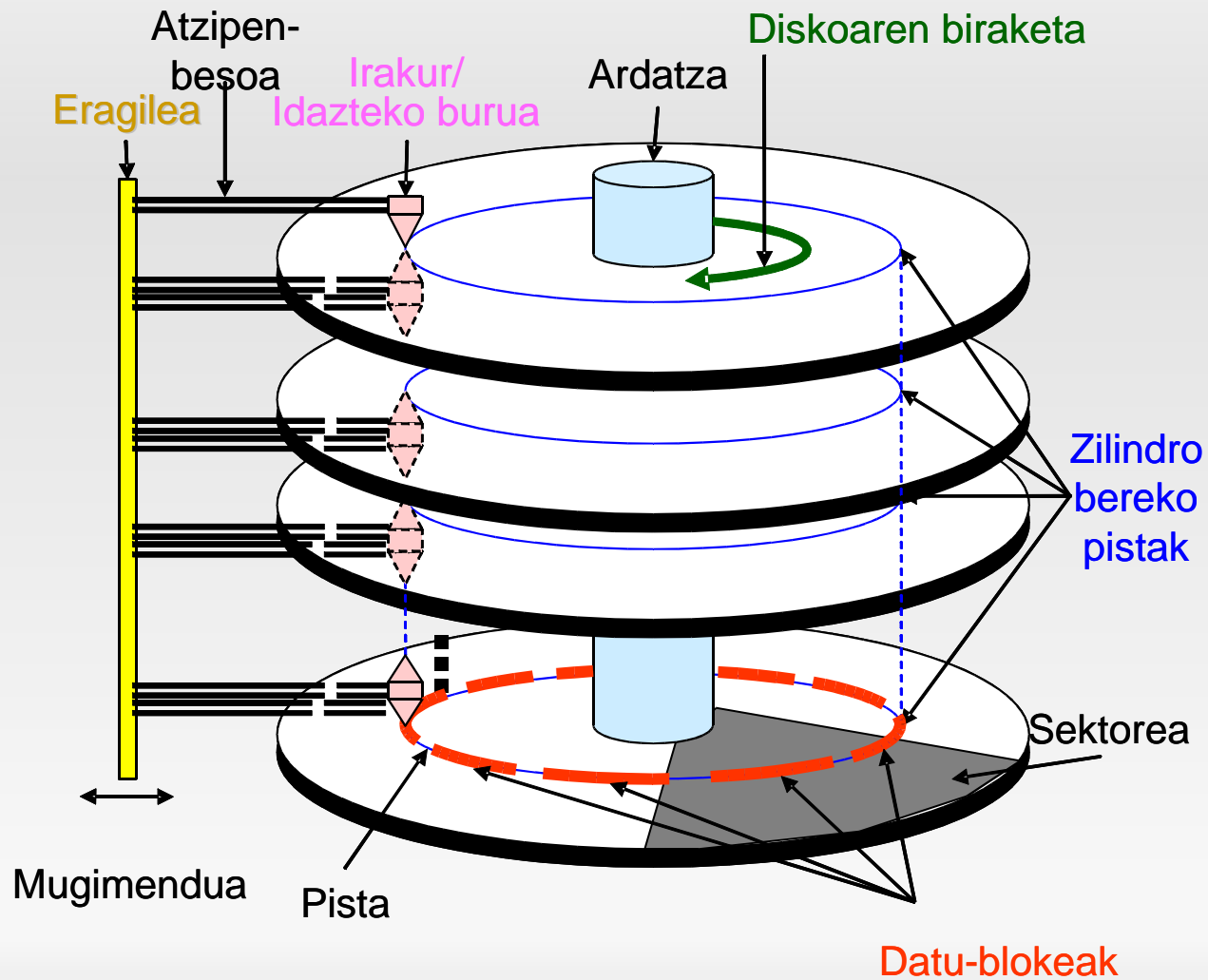
- Ordenagailuko PUZak zuzenean kudea ditzakeen gordetze-euskarriak. Bizkorrak baina gordetze-leku mugatua.
 - Memoria nagusia (gordetze hegazkorra)
 - Cache memoriak

- Bigarren eta hirugarren mailako gordetzeak

- Ordenagailuko PUZak ezin ditu zuzenean prozesatu bertako datuak; aurrena lehen mailako gordetze-euskarri batera kopiatu behar dira. Mantsoagoak baina edukiera handiagoa.
 - Diskoak eta zintak (gordetze ez-hegazkorra)
- Lehen mailako eta bigarrenen artean: flash-memoriak

DB gehienak *bigarren mailako gordetzean* biltegitratzen dira modu iraunkorrean. Diskoetan gordetako datuak [erregistroez](#) osatutako [fitxategi](#) moduan antolatzen dira.

Diskoak



Erregistroa, blokea eta bufferra

- **Erregistroa**: entitate (EE), tupla (eredu erlazionala), errenkada (SQL)...
- **Blokea**:
 - Disko eta memoria arteko **transferentzia-unitatea**
 - Erregistro bakarrarekin edo bat baino gehiagorekin osatua
- **Bloke batetara atzipen zuzena** (edo ausazkoa): helbidea gainazal-zenbakiak, pista-zenbakiak eta bloke-zenbakiak osatzen dute
- **Bufferra**: Memoria nagusiko espazioa zeinetan bloke bat kabitzen den



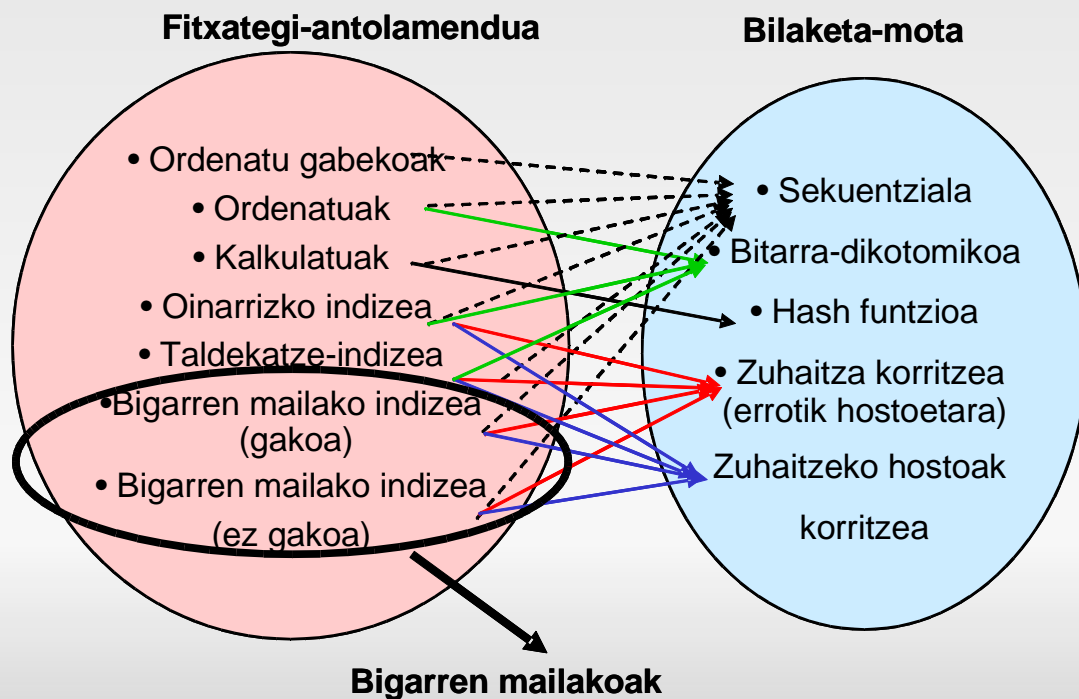
Antolamendua vs Atzipen-metodoa

- **Fitxategi-antolamendua:**

- Nola kokatzen diren erregistroak eta blokeak
- Nola lotzen diren

- **Atzipen-metodoa:**

- Nahi dugun erregistroa bilatzeko zein algoritmo erabiltzen den



- **Fitxategi-antolamendu batek atzipen-metodo bat baino gehiago onar dezake**
- Antolamendu batzuek EZ dituzte onartzen atzipen-metodo guztiak
- Kasu bakoitzean **antolamendu egokiena** bilatu behar da. Antolamendu egokiena **eragiketa ohikoenak modu eraginkorrean** egitea ahalbidetzen duena da.
- Askotan inolako antolamenduk ez du ahalbidetzen eragiketa guztiak modu eraginkorrean inplementatzea
- Espero dugunaren garrantzia eta berreskuratze- eta eguneratze-eragiketen arteko tarteko soluzio bat bilatzen da orduan



Erregistro ordenatugabeen fitxategiak (1/3)

Blokea: 1

Titulua	Urtea
Meninas	1656
Hilanderas	1644
Ronda noche	1642
Gioconda	1506
Creación de Adán	1510
Gernika	1937
El grito	1893
Pers. memoria	1924
Marilyn	1964
Girasoles	1889
Lanzas	1634
Maja desnuda	1800

2

3

4

- **Ordenatu gabeko** erregistroez osatutako fitxategiak
- Antolamendu sinpleena:
 - Erregistroak idatzi diren ordenan kokatzen dira
 - Erregistro berriak beti **bukaeran txertatzen dira**
- **Bilaketa**
 - **Lineala edo sekuentziala**: bilaketa blokez bloke jarraian egiten da baldintza betetzen duten **erregistro guztiak** bilatzeko (garestia).
 - Kasu onena: bloke **1** (lehen blokeko erregistro batek betetzen du baldintza)
 - Kasu okerrenaren kostua: **b** bloke, non b fitxategiko bloke kopurua den. Adibideak: inolako erregistrok ez duenean baldintza betetzen.
 - Tarteko kasua: **$(1+b)/2$**

Erregistro ordenatugabeen fitxategiak (2/3)

■ Txertaketa

1. Bufferrera fitxategiko azken blokea ekartzen da
2. Bufferreko blokean tokirik badago, txertatu behar den **erregistroa gehitu**. Blokean erregistroa gehitzeko tokirik ez badago, bloke berria sortzen da.
3. Bufferreko blokea diskoan **berridatzi** (berria bada, idatzi)
4. Bloke berria sortu balitz, helbidea deskribatzailean gordeko da

Hutsuneek
diskoan
espazioa
hartzen
dute

■ Ezabaketa

1. Ezabatu behar den erregistroa diskoan **bilatu**
2. Erregistroa daukan blokea memoriako **bufferrera pasa**
3. Bufferrean dagoen bloketik erregistroa **ezabatu** (edo ezabatua bezala markatu - bit batez adierazi ea erregistroa ezabatua izan den. Ezabatuak izan ez direnak erabiliko dira).
4. Bufferrean dagoen blokea dagoeneko aldatuta diskoan berridatzi

■ Espazioa galtzearen arazoa:

diskoan **hutsuneak gelditzen** dira (asko izan daitezke baldin eta ezabaketa asko egiten badira), eta beraz, espazioa galtzen da.

Erregistro ordenatugabeen fitxategiak (3/3)

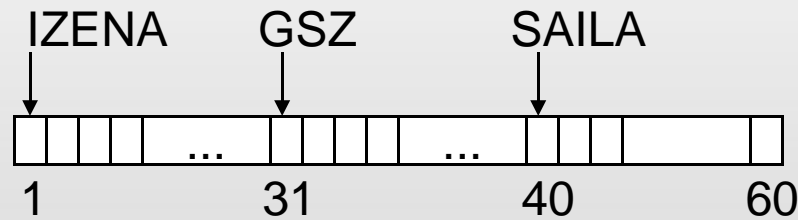
■ Ezabaketa (jarraipena):

- **1. soluzioa:** fitxategia periodikoki **berrantolatu** (erabili gabe dagoen tokia berrerabiltzeko)
 - Blokeak sekuentzialki korritzen dira
 - Erregistroak berriro kokatzen dira hutsuneak aprobetxatuz
- **2. soluzioa:** txertatzeko orduan hutsuneak aprobetxatzen saiatu eta ezin bada soilik, txertatu bukaeran
 - Kalkulu gehiago egin behar dira hutsuneak dauzkaten blokeak topatu ahal izateko (bloke asko transferitu behar dira)

■ Eguneraketa

1. Eguneratu behar den erregistroa **bilatu**
 2. Erregistroa daukan blokea **bufferrera pasa**
 3. Bufferreko blokean erregistroa **eguneratu**
 4. Bufferreko blokea diskoan **berridatzi**
 - **Blokea gainezkatzearen arazoa:** luzera aldakorreko erregistroak aldaketaren ondorioz handitzen direnean
- **Soluzioa:**
 - Eguneratu nahi den erregistroa fitxategitik ezabatu
 - Erregistro eguneratua fitxategian txertatu

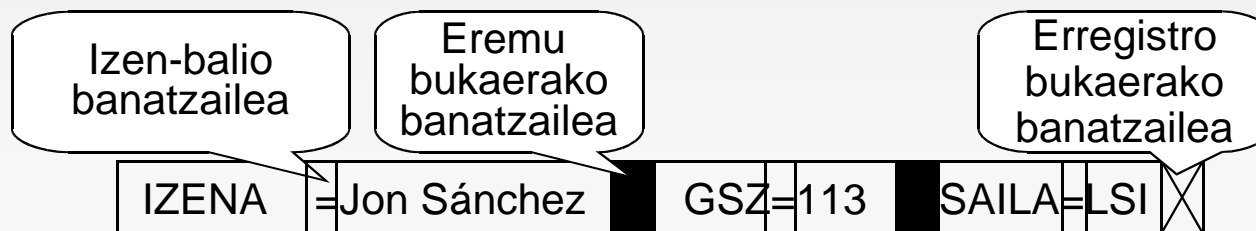
Erregistroak: luzera finkoa edo aldakorra



Luzera finkoko erregistroa:
3 eremu eta 60 byte



Luzera finkoko 2 eremu (GSZ eta SAILA) eta luzera aldakorreko 2 eremu (IZENA eta SOLDATA) dituen erregistroa



Luzera aldakorreko eremuak eta mota ezberdinetako karaktere banatzaileak dituen erregistroa

Erregistro ordenatuen fitxategiak

Ordenatutako fitxategi bateko blokeak.
Ordenazio-eremutzat titulua dute.

Blokeak: 1

Titulua	Urt
Creación Adán	1510
El grito	1893
Gioconda	1506

2

Girasoles	1889
Guernica	1937
Hilanderas	1644

3

Lanzas	1634
Maja desnuda	1800
Marilyn	1964

4

Meninas	1656
Pers. memoria	1924
Ronda noche	1642

- Ez da ordenatu berri den erregistro ordenatugabeen fitxategia
- Erregistroak, fisikoki eremu berezi baten balioen arabera ordenatuta **mantentzen dira: ordenazio-eremuaren** arabera
- Ordenazio-eremua gakoa baldin bada, **ordenazio-gakoa** deitzen zaio
- Ordenazio-gakoa, **hainbat eremuren** konbinazioa izan daiteke

Zbkia	Let	Ize	Abi
1	a	jon	perez
1	e	feli	arrieta
2	b	ander	azpeitia
2	d	ana	martin
2	z	libe	reverte
3	j	juan	imaz
5	a	amaia	sastre
9	a	iker	arregi
9	h	irati	zurbano

Ordenatutako fitxategi bateko blokeak.
Ordenazio-eremua **Zbkia + Let**-en konbinazioa da (ordena horretan)

Fitxategia **Zbkia**-ren arabera ordenatuta mantentzen da baina ez **Let**-en arabera.

Erregistro ordenatuen fitxategiak: abantailak

- Erregistroak **ordenazio-eremuaren ordenaren** arabera modu eraginkorrean lortzen ditu (nahikoa da sekuentzian irakurtzea)
- **Ordenazio-eremuaren ordenan** hurrengoa den erregistroaren atzipen azkarra
 - Buffer berberean dago (irakurtzeke erregistroak baldin badaude) edo,
 - bestela, hurrengo blokea ekarri (baldin badago)
- **Bilaketa bitarra** egiteko aukera:
 - Askoz azkarragoa
 - Soilik erabil daiteke, bilaketa egiterakoan aztertuko den **baldintza ordenazio-eremuan** oinarrituta baldin badago
- Fitxategi hauek **EZ dute inolako abantailarik eskaintzen**, bilaketak **ordenazio-eremua** ez den eremuren baten arabera egiten direnean

Bilaketa bitarra (dikotomikoa)

- Blokeen gainean egiten da
- Adibidea: 1, 2, ..., b blokeak
- 1, 2, ..., b blokeen helbideak deskribatzailean daude
- **Ordenazio-gakoan** k balioa duen erregistro **bat** topatu nahi da. Bataz besteko kostua: **$\log_2(b)$** bloke. Bataz besteko kostua **bilaketa linealarekin**: b/2 (b topatu nahi den erregistroa ez badago)
- K balioa duten erregistro **batzuk** topatu ahal dira **gakoa ez den ordenazio-eremua** erabilita
- Adibidez, <, >, <= eta >= erabiliz bilaketa eginez

Erregistro ordenatuen fitxategiak: eragiketak

(1/2)

- **Ordenazio-eremuaren** arabera erregistroen ordena mantendu behar denez, **txertaketa** eta **ezabaketa** **garestiak** dira

Txertaketa

1. Erregistroa ordenan txertatzeko bloke egokia **bilatu** behar da (bilaketa bitarra erabiliz)
2. Blokea bufferrera **transferitzen** da
3. **Hutsune bat** egin behar da bufferreko blokean, horretarako, blokeko gainontzeko erregistroak **desplazatu** egin behar dira
4. Erregistro berria bufferreko blokean **txertatu** eta blokea diskoan **berridatzi**
5. Blokean tokirik ez balego, soberan dagoen erregistroa (**ordenazio-eremuan** balio handiena duena) hurrengo bloke mugitu behar da. Kasurik okerrean honek fitxategiko bloke guztiei eragiten die.

- Beste aukera batzuk:
 - **Hutsuneak** utzi bloketan: betetzen direnean arazo berbera edukiko dugu
 - **Gainezkatze-fitxategia** (*overflow*):
 - Erregistroak EZ DAUDE ordenatuta
 - Erregistro berriak bukaeran txertatzen dira
 - Aldizka fitxategi ordenatua berrantolatu behar da: ordenatutako fitxategia berreraiki behar da bere posizioan gainezkatze-fitxategiko erregistroak ipiniaz
 - Horrela, txertatze-prozesua arindu egiten da
 - Bilaketa-prozesua motelagoa da

Erregistro ordenatuen : eragiketak (2/2)

Ezabaketa

- Fisikoki ezabatu nahi bada, txertaketa-prozesuaren tankerakoa
- Sinpleagoa da erregistroa ezabatua bezala markatzen baldin bada eta berrantolaketa periodikoa egiten bada

Eguneraketa

- Kasuaren arabera bilaketa bitarra egiteko aukera
- Eguneratzen den eremua **ordenazio-eremua** baldin bada honek posizioa aldatzea suposa lezake. Erregistroa ezabatzea eta berridaztea beharrezkoa izan daiteke
- Erregistroak luzera finkokoak baldin badira, blokea toki berean berridatzi ahalko da

DBetan fitxategi ordenatuen erabilera:

- DBen aplikazioetan gutxi erabiltzen dira hemen adierazitako moduan
- Ohikoagoa da indize batekin batera erabiltzea. Honi **oinarrizko-indize** deitzen zaio (**ordenazio-gakoak** erabiltzen direnean, adibidez, tauletako **oinarrizko gakoak**) edo **taldekatze-indizea** (**ordenazio-eremua** gakoa ez denean)

Hashing teknikak

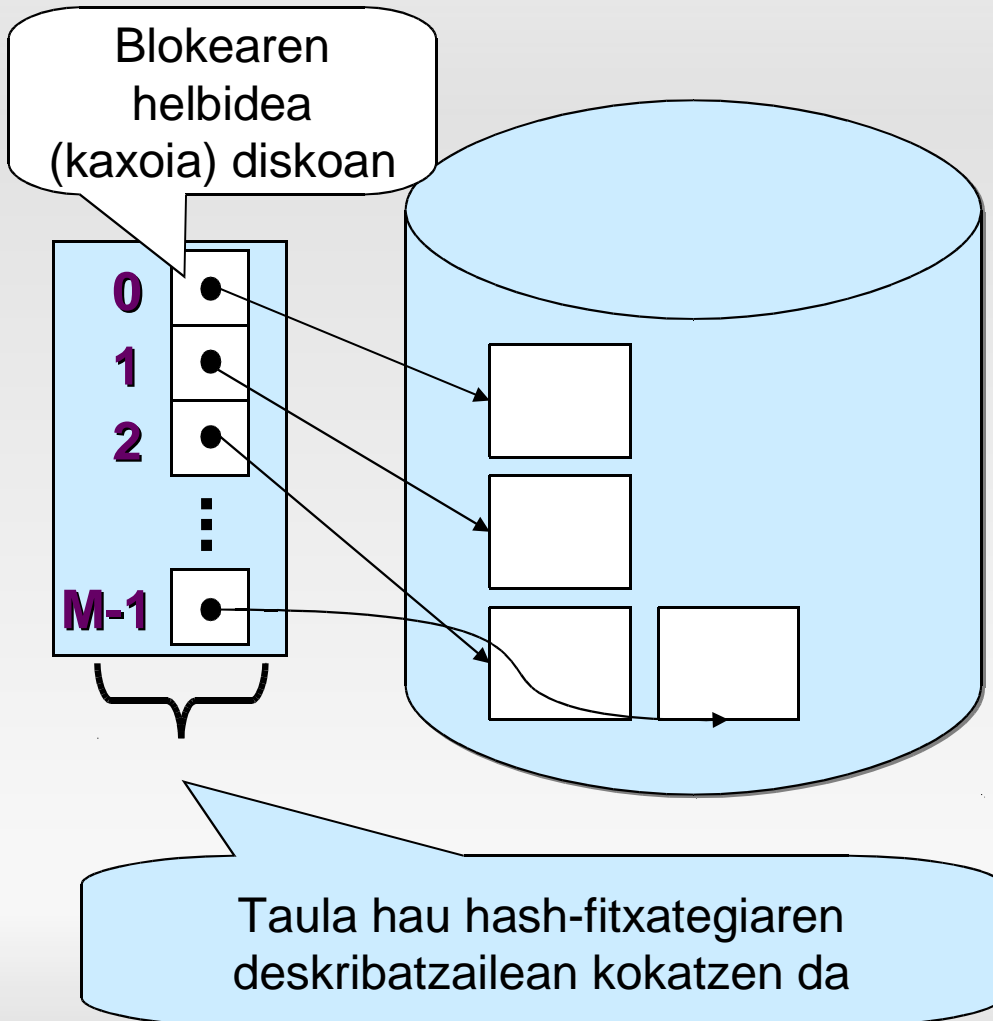
- *Zuzeneko fitxategi* ere deitu izan zaie
- Zenbait bilaketetarako **atzipen azkarragoa** (orokorrean, bloke bakarra transferitzen da)
 - Bilatze-baldintza: *eremu bakarrean ezartzen den berdintasun-baldintza* (=)
- Kalkulatuko eremu hau gehienetan **gakoa** da (hash-gakoa)
- Aleatorizazio-funtzioa edo hash-funtzio
 - Hash-eremuko balioari aplikatzen zaion funtzioa

- Eraitza: erregistroa gordeta dagoen disko-blokeko helbidea
 - Bloke horretan bilatutako erregistroa dago
 - Erregistroa bufferrean bilatzen da
 - Bufferrera atzipen bakarra nahikoa da
- *Barne-hashing-a*
 - Programazioan oso erabilgarria den datu-egitura (barne-memorian)
 - *Kanpo-hashing-a*
 - Disko-fitxategiekin hashing-a (hau baino ez dugu ikusiko)



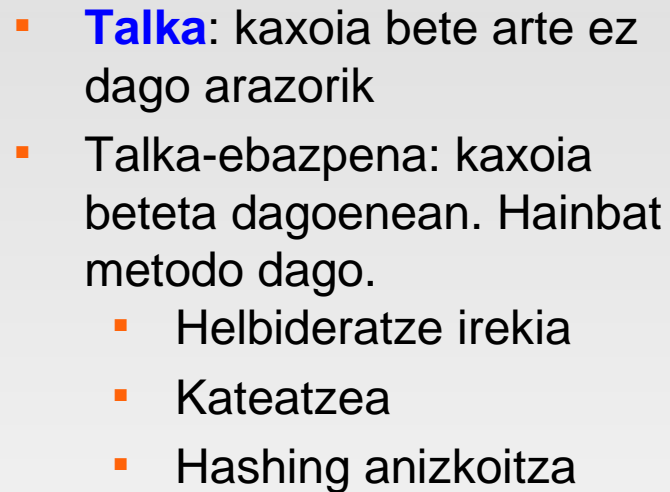
Kanpo-hashing-a (1/2)

Kaxoi-zenbakiaren eta diskoan blokeak duen helbidearen arteko harremana:



- **Hash-funtzioa**
 - Gako bat kaxoi-zenbaki erlatiboarekin mapatzen du
 - Argumentua: **eremuaren** balioa
 - **0** eta **M-1** arteko balio bat ematen du bueltan (taulan posizioa)
 - Adibidez, $\text{hashF}(X) = x \bmod M$
- Taulako posizio horretan **kaxoi** baten helbidea dago
- **Kaxoi** bat bloke batekin edo bloke jarrai sekuentzia batekin osatuta dago
- Taula diskoan gordetzen da, deskribatzaileari dagokion blokean. Taula erabiliz, kaxoiaren zenbaki erlatiboa, bloke-helbide bihurtzen da.

Kanpo-hashing-a (2/2)



Zer ematen du fhash(321)-ek?
Eta fhash(580)-ek?
Eta fhash(981)-ek?
Eta fhash(652)-ek?

Kanpo-hasing-a: arazoak eta eragiketak

■ Arazoak:

- Hash-funtzioek orokorrean EZ dute gakoaren ordena mantentzen
- **Aldez aurretik zehaztutako espazio kantitatea:**
 - M kaxoi * m erregistro/kaxoi
 - Erregistro kopurua askoz txikiagoa bada: espazioa galtzen da
 - Askoz ere handiagoa bada:
 - Gatazka asko
 - Gainezkatze-kaxoi zerrenda luzeak
 - Bi kasutan hobe da:
 - Esleitutako espazioa aldatzea
 - Hash-funtzioa aldatzea
 - Erregistroak berkokatzea

■ Eragiketak

■ Bilaketa:

- Eremua \neq **helbideratze-eremua** denean, ordenatu gabeko fitxategi batean bezain garestia

■ Ezabaketa:

- Kaxoian badago: ezabatu eta gainezkatze areatik erregistro bat pasa
- Gainezkatze arean badago: ezabatu eta zerrenda estekatua eguneratu

■ Eguneraketa: Bi kasu daude:

- **Hash-eremua:** ezabatu eta txertatu
- Beste eremu bat: berridazketa

Hedapen dinamikoa onartzen duten hashing-teknikak

- Fitxategiaren tamaina dinamikoki aldatzen da beharren arabera
- Helbideen espazio finkoaren arazoa konpontzen du
- Hashing hedagarria:
 - Direktorioa + kaxoiak

Direktorioa —————>

00	01	10	11				
000	001	010	011	100	101	110	111

- Helbideratze kalkulatu lineala
- Hainbat hash funtzioekin

(ez ditugu ikusiko)

5. gaia: Diseinu fisikoa

- *Helburua*
- *Datu-baseen gordetzea eta fitxategi-antolamendua*
Elmasri & Navathe 09
- *Fitxategietako indexatze-egiturak edo atzipen-egiturak*
Elmasri & Navathe 09
- *Diseinu fisikorako gomendioak eta diseinua nola egin*
Conolly & Begg 05
- *Kontsulta-optimizazio*
Elmasri & Navathe 09



Atzipen-egiturak. Indizeak

Indexatze-

eremua

Erakuslea

Aaron

Adams

Alex

Angel

.....

Bloque: 1

2

3

4

Título	DNI
Aaron	1510
	1893
Acosta	1506
Adams	1889
	1937
Akerra	1644
Alex	1634
	1800
Anderson	1964
Angel	1656
	1924
Arnold	1642

- Indizeek indexatze-eremuaren balioak gordetzen dituzte erakusleekin batera
- Indizean indexatze-eremuaren balioak ordenatuta daude
- Indize trinko batek sarrera bat du erregistro bakoitzeko. Indize ez-trinkoak sarrera bat du bloke bakoitzeko.

Indizeak

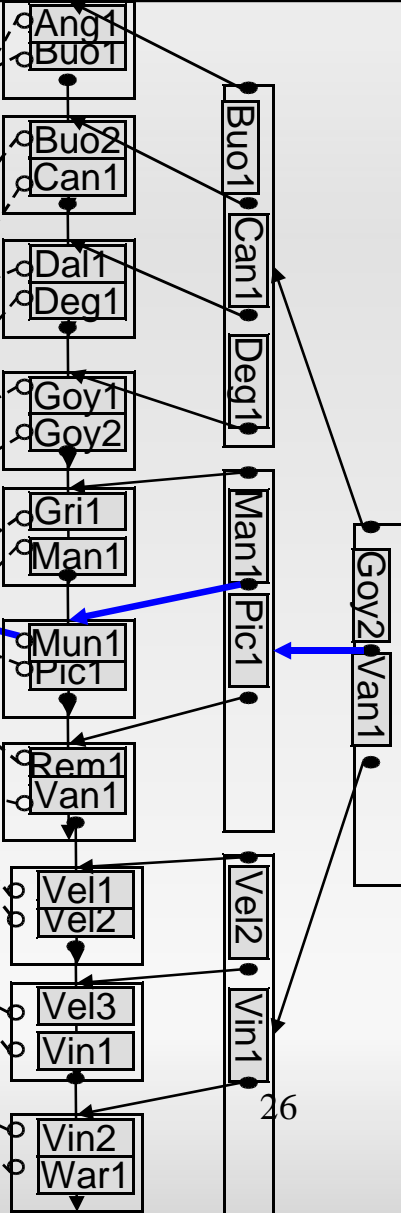
- Erregistroen atzipena azkartzen duten **atzipen-egiturak** dira (bilaketa-baldintza batzuetan behintzat)
- DBKS gehienek **B+ zuhaitzekin** inplementatzen dituzte beraien indizeak (hauek soilik ikusiko ditugu)
- Lau indize mota bereizten dira:
 - **Oinarrizko indizeak** (gakoa)
 - **Multzokatze-indizeak** (ez gakoa)
 - **Bigarren mailako indizeak** gakoaren gainean
 - **Bigarren mailako indizeak** gako ez denaren gainean
- Indize mota ezberdinak bereizten dira: batzuk fisikoak (erakusleekin) eta besteak logikoak (oinarrizko gakoan balioak dituztenak, erakusleen ordeaz)
- Indize bat fitxategiko **eremu bakarraren** edo **eremu multzo** baten gainean eraiki daiteke
- Fitxategi batek indize bat baino gehiago onartzen ditu (*kontuz nolakoak!*)
- **Oinarrizko indizeek** eta **multzokatze-indizeek** fitxategiaren antolamendua **ordenatua** izatea eskatzen dute
- Fitxategi batek indize asko eduki ditzake baina **oinarrizko** indize edo **multzokatze-indize bakarra** eduki dezake
 - Gainontzekoek bigarren mailakoak izan beharko dute
- **Bigarren mailako indizeek** edozein fitxategi-antolamendu onartzen dute

Gakoaren gainerako bigarren mailako indizea (B+ zuhaitza)

INDIZEA
ID eremuaren
gainera
(B+ zuhaitza)

DATU-fitxategia

ID	Titulua	Urt	Margolaria	Estiloa	Tamaina
Vel1	Meninas	1656	Velazquez	Barrokoa	318x276
Vel2	Hilanderas	1644	Velazquez	Barrokoa	222x293
Rem1	Ronda noche	1642	Rembrandt	Barrokoa	363x437
Vin1	Gioconda	1506	Da Vinci	Errenazimentu	077x053
Buo1	Creación Adán	1510	Buonarroti	Errenazimentu	null
Pic1	Guernica	1937	Picasso	Kubismoa	349x776
Mun1	El grito	1893	Munch	Espresionismoa	091x073
Dal1	Pers. memoria	1924	Dalí	Surrealismoa	024x033
War1	Marilyn	1964	Warhol	Pop Art	101x101
Van1	Girasoles	1889	Van Gogh	Inpresionismoa	095x073
Vel3	Lanzas	1634	Velazquez	Barrokoa	307x367
Goy1	Maja desnuda	1800	Goya	null	097x190
Goy2	Aquelarre	1823	Goya	null	140x435
Vin2	Última cena	1497	Da Vinci	Errenazimentu	480x880
Buo2	Juicio final	1541	Buonarroti	Errenazimentu	1370x1220
Gri1	Guitarra	1913	Gris	Kubismoa	061x050
Deg1	Cabaret	1877	Degas	Inpresionismoa	024x044
Ang1	Anunciación	1434	Angelico,F.	Gotikoa	175x080
Can1	Pz. S. Marco	1750	Canaletto	Barrokoa	068x112
Man1	El flautista	1866	Manet	Inpresionismoa	161x097



Blokea: 1

2

3

4

5

6

7

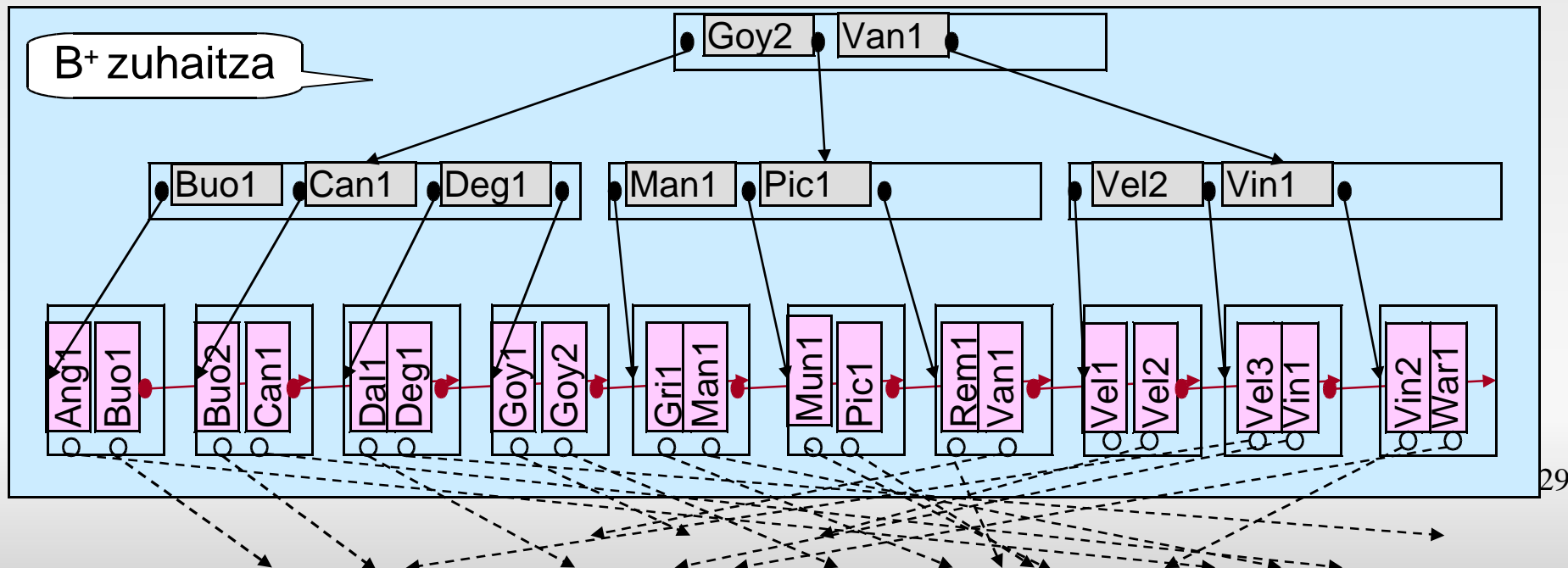
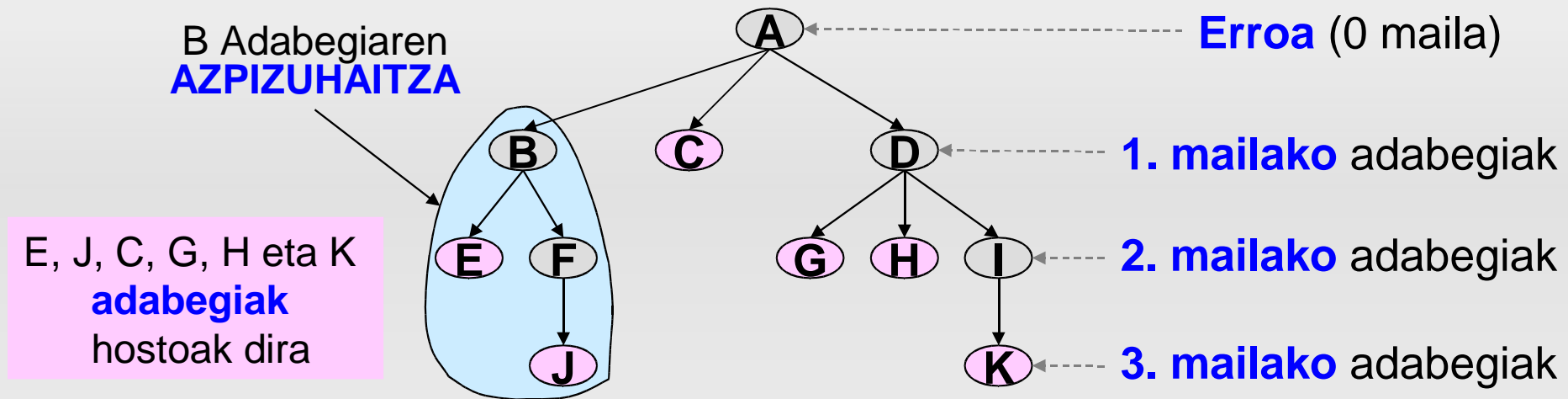
Gakoaren gainerako bigarren mailako indizea (B+ zuhaitza) (2/2)

- Bigarren mailako indizea gako eremuaren gainean (Bi-g):
 - Adibidean, *ID* eremuarentzako definitu da
 - Hostoak: sarrera bat, datu-fitxategiko erregistro bakoitzarentzako. Sarrera bakoitzak, IDren balio bat eta balio hori duen erregistrora erakusle bat ditu. Sarrerak hostoetan ordenatuak daude. Hostoak haien artean estekatuak daude, eta ezkerretik eskuinerantz irakur daitezke.
- Datu-fitxategia: Edozein oinarritzko antolamendu izan dezake (ordenatu gabea, ordenatua, helbideratze kalkulatua e.a.). Beraz, fitxategi batentzako Bi bat baino gehiago defini daitezke.
- Ordenazio logikoa: Bi-g erabiliz, erregistroak ID-ren araberako goranzko ordenan irakur daitezke. Irakurketa horretarako, bloke-transferentzia asko behar izatea gerta daiteke (kasurik okerrenean: erregistro adina)
- ID= "Mun1" duen erregistroaren bilaketa: errotik hasita, erdiko erakuslea jarraitzen da (x-ren "**Goy2**" < x ≤ "**Van1**" balioak). Erdiko-mailan, erdiko erakuslea jarraitzen da (honek x-ren "**Man1**" < x ≤ "**Pic1**" balioetara eramaten gaitu). Hostoan, bilatutako "**Mun1**" balioa dago (egongo ez balitz, existitzen ez delako izango da). "**Mun1**" balioaren erakuslea jarraitzen da, ID="Mun1" balioa duten erregistro guztiak lortzen dira (kasu honetan gakoa izanik bakarra).
- Bilaketa guztietan maila bakoitzeko adabegi bat korritzen da (bloke-transferentzia bat egiten da adabegi bakoitzeko) eta gero, beste transferentzia bat egiten da datu-erregistroa duen blokea lortzeko
- Erregistroak txertatzea/ezabatzea: Indizearen eguneratzea eta berrantolatzea eskatzen du

Maila anitzeko indize dinamikoak B zuhaitzak erabiliz

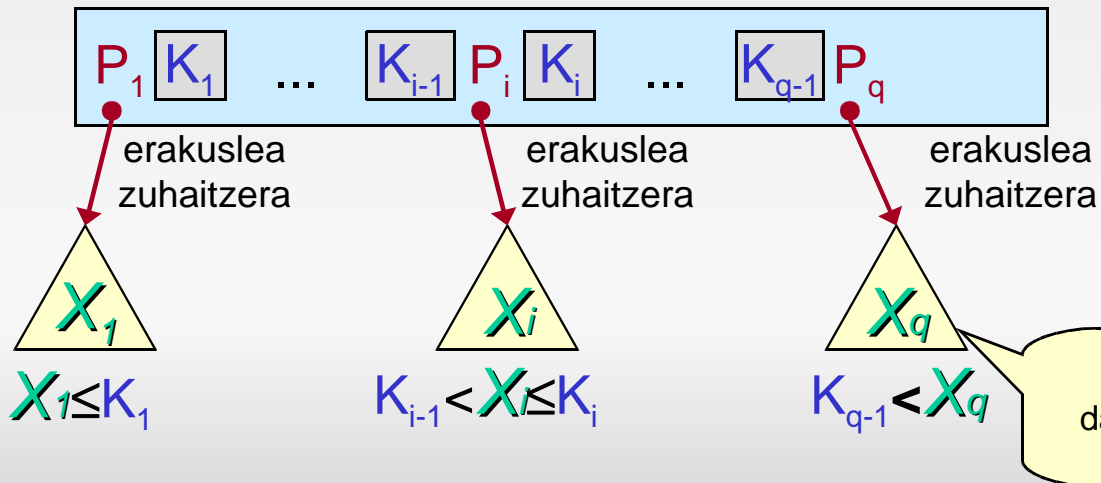
- B eta B+ zuhaitzen kasu partikularrak dira (zuhaitz orekatuak)
 - Zuhaitz bat adabegiekin osatuta dago
 - Zuhaitzeko adabegi bakoitzak (erroak izan ezik) guraso-adabegi bat du eta hainbat ume-adabegi
 - Umerik gabeko adabegia hostoa da
 - Hostoa ez den adabegia, barne-adabegia da
 - Adabegi baten maila, beti bere gurasoarena baina bat gehiago da (0 maila erroarena da)
 - Adabegi baten azpizuhaitza adabegi horrekin eta bere ondorengo guztiekin osatuta dago
 - B+ zuhaitzetan fitxategi bateko datu-blokeetara erakusleak hostoetan soilik biltegiratzen dira

Zuhaitzei buruzko terminologia eta B⁺ zuhaitzaren adibidea



B+ zuhaitzak (1/2)

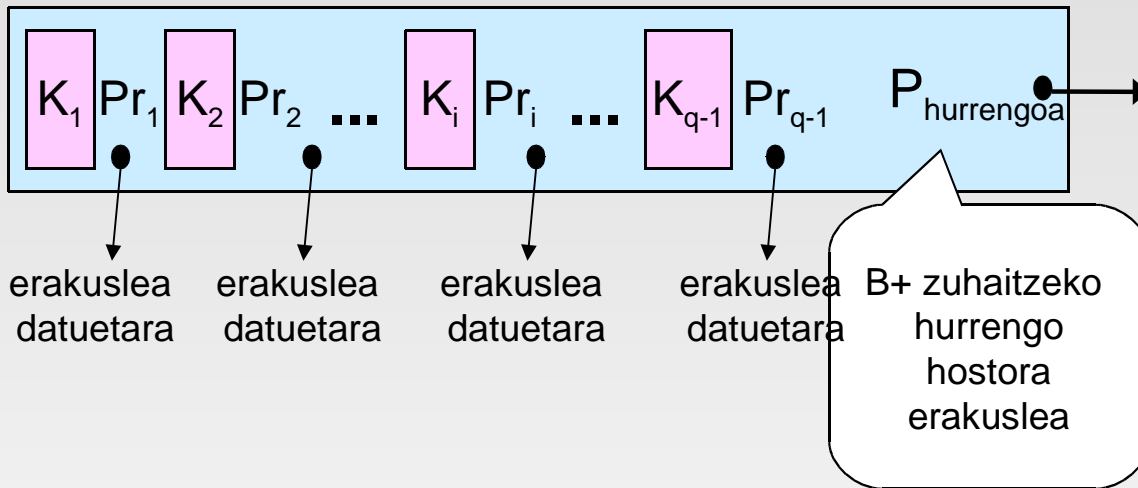
- Beti orekatua
- Espazioa alferrik galtzen da baina nahiko modu onargarrian
- Txertaketa eta ezabaketa konplexuak
- **Barne-adabegiak** eta **hostoak** desberdinak dira
- **Barne-adabegietan** balioak (K_i -rekin adieraziak) eta beste adabegietara erakusleak (P_i) daude:



- **Barne-adabegien** ezaugarriak:
 - Adabegi batean K_i ordenatuak:
 $K_1 < K_2 < \dots < K_{q-1}$
 - Adabegi guztiek tamaina bera: p ordenakoa bada gehienez p erakusle eta $p-1$ balio desberdin
 - Adabegi bakoitzean gutxienez $\lceil p/2 \rceil$ erakusle daude beteta. Erroak gutxienez 2 erakusle ditu (barne-adabegia denean, hostoa izan baitaiteke)
- K_i bakoitza **errepikatu egiten da** bere ezkerrera dagoen azpizuhaitzeko hosto batean (azpizuhaitz horretako balio altuena izango da)

B+ zuhaitzak (2/2)

- **Hostoetan** balioak (K_i) eta datu-blokeetara erakusleak (Pr_i) daude:



- Indexatze-eremua (K_i balioak dauzkana) **ez** bada gakoa: helbideratze-maila gehigarri bat erabiltzen da (erakusleen blokeetara).

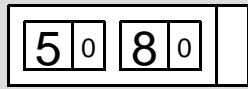
- **Hostoen** ezaugarriak:

- Hosto batean $K_1 < K_2 < \dots < K_{q-1}$ eta $q \leq p$ (ordenatuak)
- Adabegi guztiek tamaina bera: gehienez p erakusle eta p K_i balio
- Pr_i -ak K_i gakoetako erregistroetara erakusleak dira (edo erregistroa duen blokerak edo erakusleak dauzkan blokerak)
- Hosto bakoitzean gutxienez $\lceil P/2 \rceil$ erakusle (eta balio)
- Hosto guztiak maila berean
- Hostoak beraien artean estekatuta: erregistroak K_i -ren goranzko ordenan korritu daitezke
- Hosto batzuen K_i balioak barne-adabegietan errepikatzen dira
- B+ zuhaitz batzuetan, erakusleak daude aurreko hostoetara

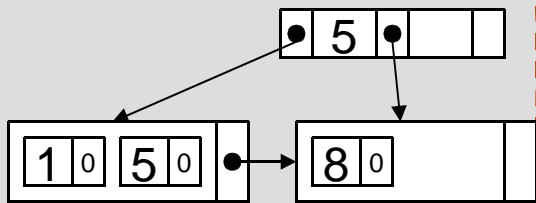
B+ zuhaitzak: txertatzea

Erakusleak:
● adabegietara
□ datuetara
□ Null zuhaitzetara

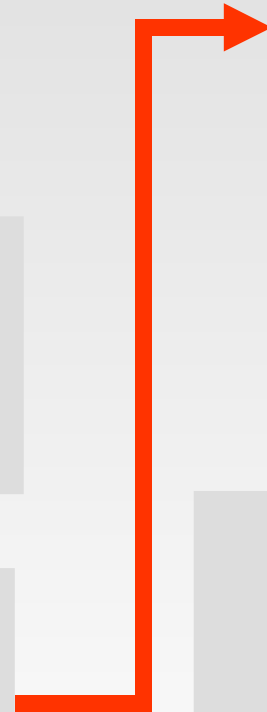
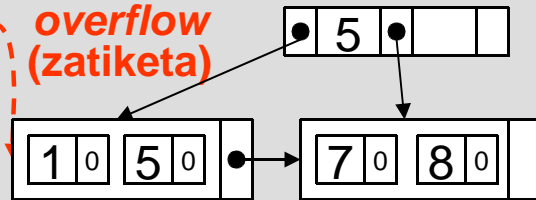
1 txertatu: **overflow**
(zatiketa eta maila berria)



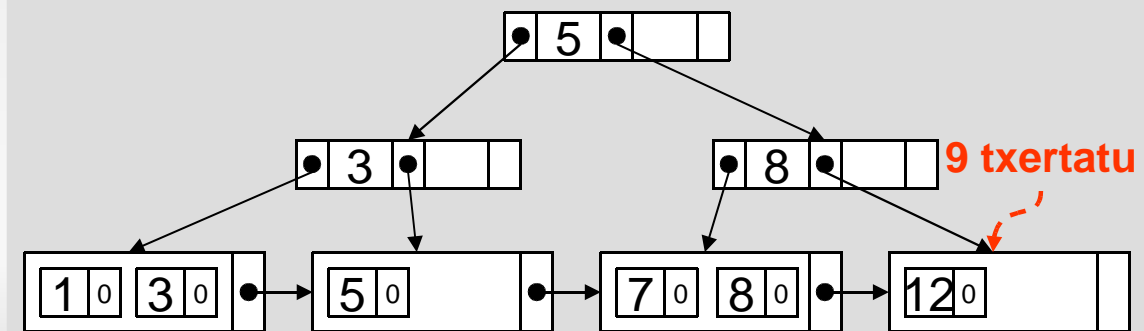
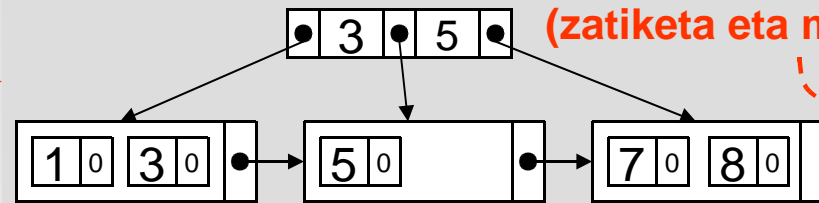
7 txertatu



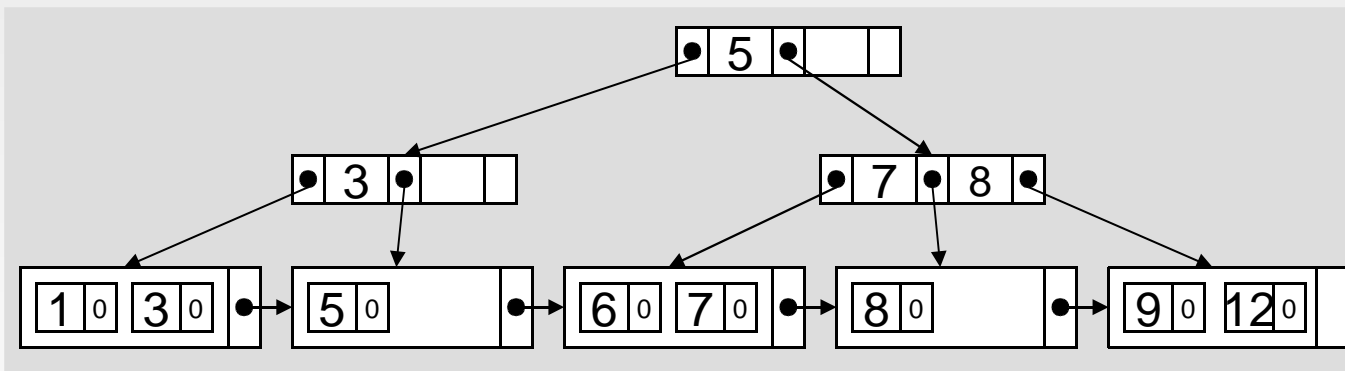
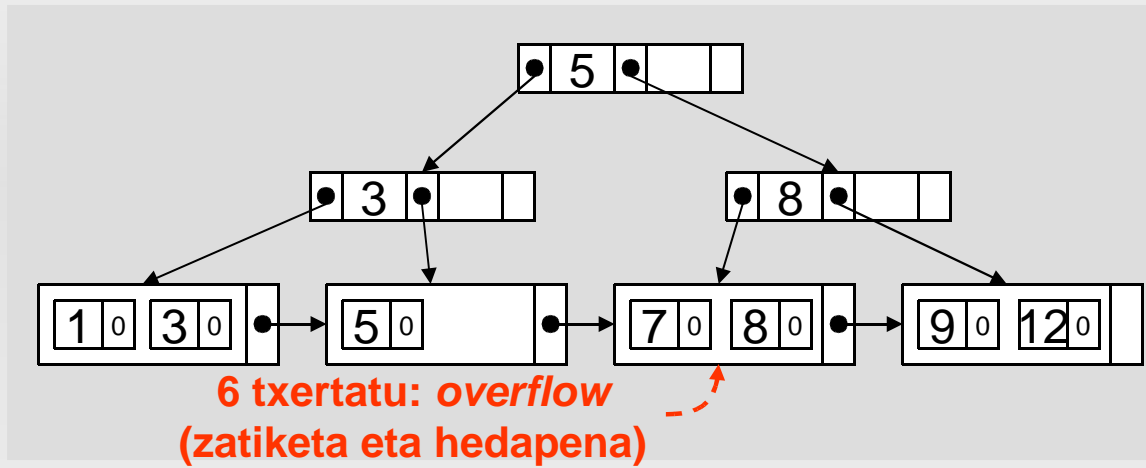
3 txertatu:
overflow
(zatiketa)



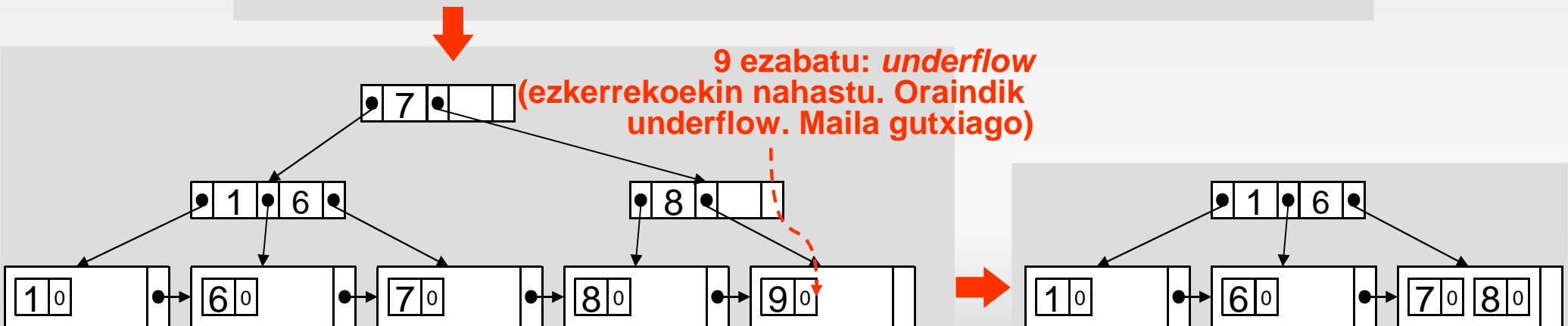
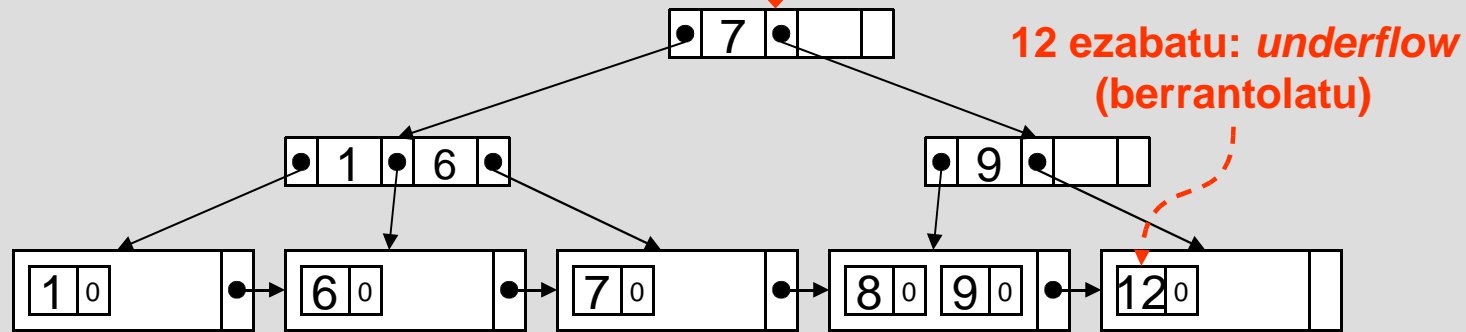
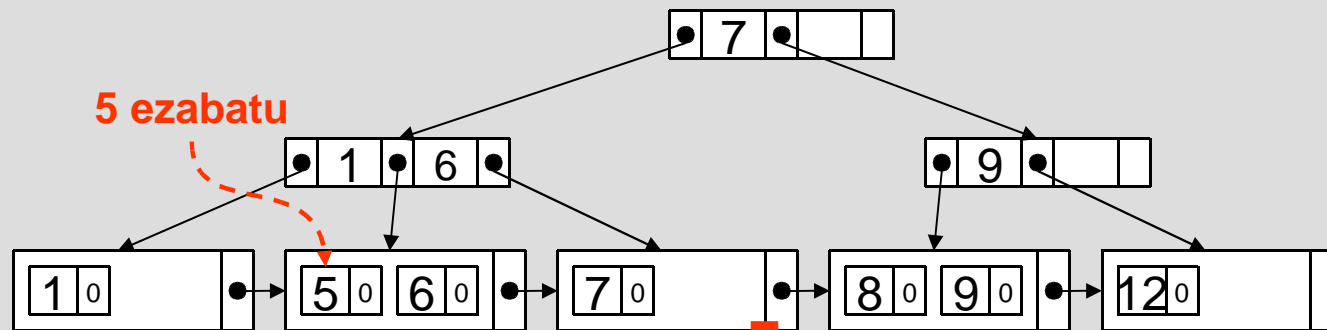
12 txertatu: **overflow**
(zatiketa eta maila berria)



B+ zuhaitzak: txertaketak (2)



B+ zuhaitzak: ezabaketak



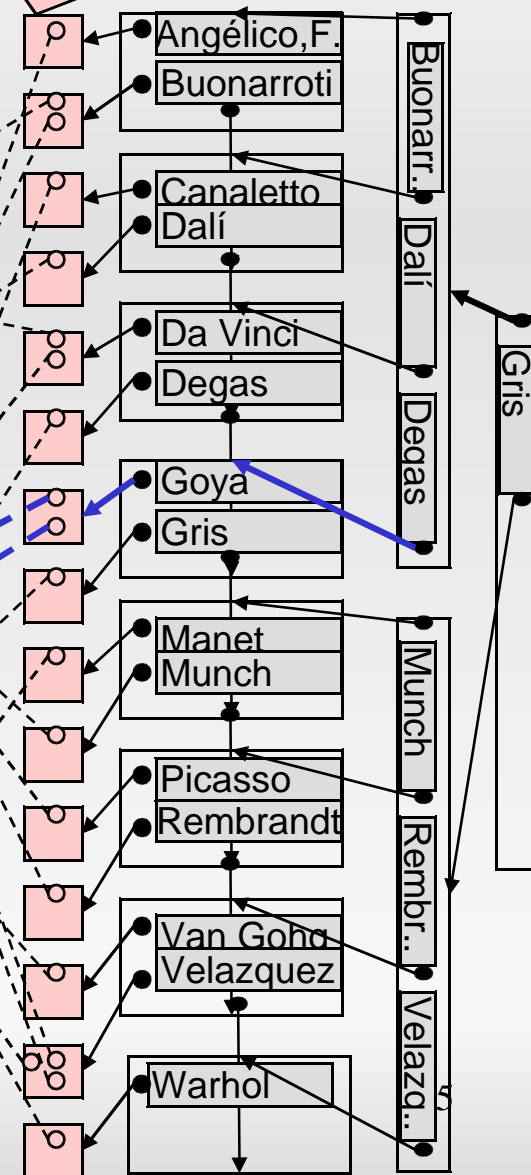
Gakoa EZ den eremuaren gaineko bigarren mailako indizea (B+ zuhaitza) (1/2)

INDIZEA
Margolaria
eremuaren
gainean
(B+ zuhaitza)

DATU-fitxategia

Erregistro-blokeetara erakusleak

ID	Titulua	Urt	Margolaria	Estiloa	Tamaina
Vel1	Meninas	1656	Velazquez	Barrokoa	318x276
Vel2	Hilanderas	1644	Velazquez	Barrokoa	222x293
Rem1	Ronda noche	1642	Rembrandt	Barrokoa	363x437
Vin1	Gioconda	1506	Da Vinci	Errenazimentu	077x053
Buo1	Creación Adán	1510	Buonarroti	Errenazimentu	null
Pic1	Guernica	1937	Picasso	Kubismoa	349x776
Mun1	El grito	1893	Munch	Espresionismoa	091x073
Dal1	Pers. memoria	1924	Dalí	Surrealismoa	024x033
War1	Marilyn	1964	Warhol	Pop Art	101x101
Van1	Girasoles	1889	Van Gogh	Inpresionismoa	095x073
Vel3	Lanzas	1634	Velazquez	Barrokoa	307x367
Goy1	Maja desnuda	1800	Goya	null	097x190
Goy2	Aquelarre	1823	Goya	null	140x435
Vin2	Última cena	1497	Da Vinci	Errenazimentu	480x880
Buo2	Juicio final	1541	Buonarroti	Errenazimentu	1370x1220
Gri1	Guitarra	1913	Gris	Kubismoa	061x050
Deg1	Cabaret	1877	Degas	Inpresionismoa	024x044
Ang1	Anunciación	1434	Angelico,F.	Gotikoa	175x080
Can1	Pz. S. Marco	1750	Canaletto	Barrokoa	068x112
Man1	El flautista	1866	Manet	Inpresionismoa	161x097



Gakoa EZ den eremuaren gaineko bigarren mailako indizea (B+ zuhaitza)(2/2)

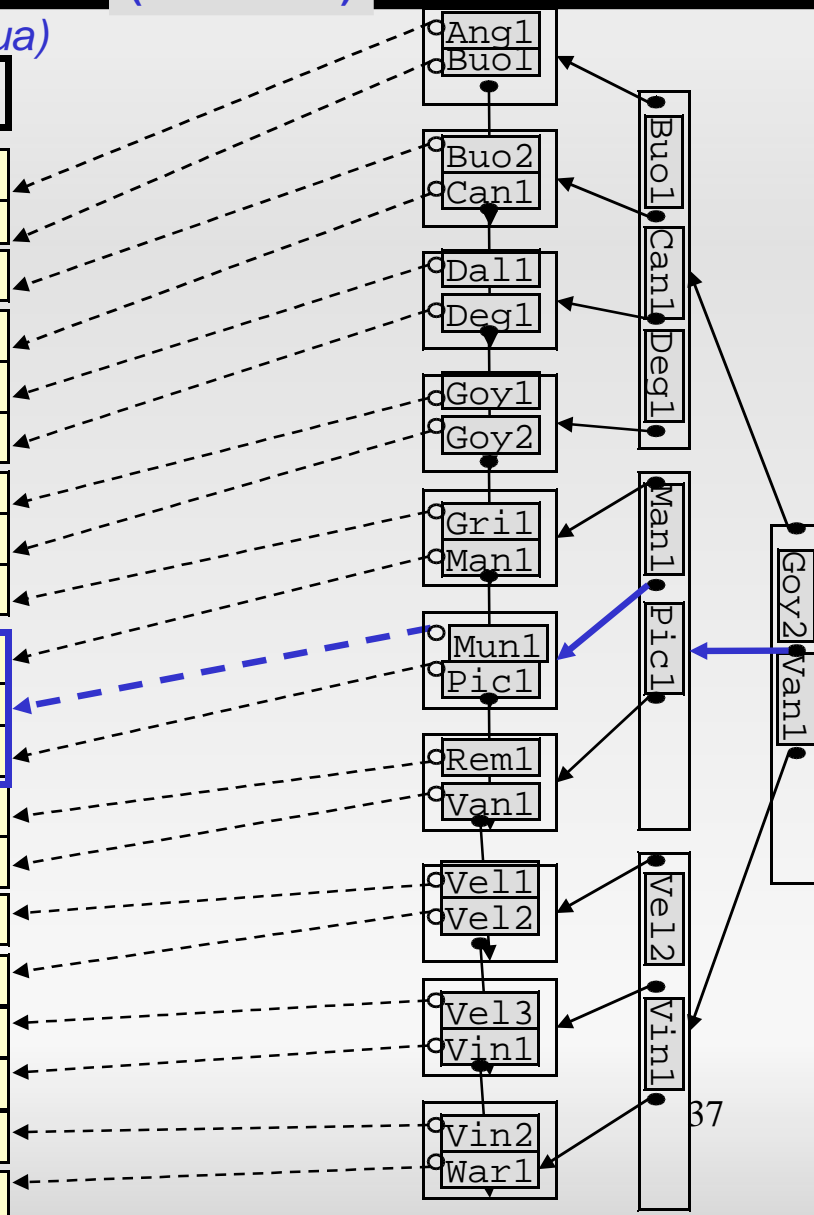
- Bigarren mailako indizea gakoa ez den eremu(ar)en gainean (Bi-eg):
 - Adibidean, *Margolaria* eremuarentzako definitu da
 - Bi-g indizearekin diferentzia bat du: hostoetako erakusleek indexatze-eremuko balio bera duten erregistro bat baino gehiago erakuts ditzakete. Horretarako, **erregistro-blokeetara erakusleak dituen helbideratze maila** bat gehiago dago (helbideratze maila honen ordeaz, beste aukera batzuk daude).
- Datu-fitxategia: Bi-g eta Bi-eg indizeekin edozein oinarritzko antolamendu izan dezakegu (ordenatu gabea, ordenatua, helbideratze kalkulatua e.a.). Beraz, fitxategi batentzako bigarren mailako (Bi-g edo Bi-eg) bat baino gehiago defini daitezke.
- Ordenazio logikoa: edo erregistroak indexatze-eremu(ar)en ordenan irakurri ahal izatea. Bigarren mailako indizeentzako soilik (oinarritzko eta taldekatze-indizeek aukera hobea dute).
- Erregistro baten bilatzea: Bi-g indizeekin bezala, baina kasu honetan erakusleen blokeak dituen maila bisitatu behar da.
- Bilaketa guztietan maila bakoitzeko adabegi bat korritzen da (bloke-transferentzia bat egiten da adabegi bakoitzeko), gero erakusleak dituen bloke bat (edo erakusle guztiak bloke bakarrean kabitzen ez badira, batzuk), eta azkenik, transferentzia bat egiten da bilatzen den balioa duen datu-erregistro bakoitzeko (batzuk egon daitezke, ez baita gakoa).
- Erregistroak txertatzea/ezabatzea: edozein indizeetan bezala, indizearen eguneratzea eta berrantolatzea eskatzen du

Oinarrizko indizea B⁺ zuhaitzarekin (oinarrizko gakoa) (1/2)

INDIZEA
Oinarrizko
indizearen
gainean
(ID)
(B⁺ zuhaitza)

DATU-fitxategia (oinarrizko gakoaren arabera ordenatua)

ID	Titulua	Urt	Margolaria	Estiloa	Tamaina
Ang1	Anunciación	1434	Angelico, F.	Gotikoa	175x080
Buo1	Creación Adán	1510	Buonarroti	Errenazimentu	null
Buo2	Juicio final	1541	Buonarroti	Errenazimentu	1370x1220
Can1	Pz. S. Marco	1750	Canaletto	Barrokoa	068x112
Dal1	Pers. memoria	1924	Dalí	Surrealismoa	024x033
Deg1	Cabaret	1877	Degas	Inpresionismoa	024x044
Goy1	Maja desnuda	1800	Goya	null	097x190
Goy2	Aquelarre	1823	Goya	null	140x435
Gri1	Guitarra	1913	Gris	Kubismoa	061x050
Man1	El flautista	1866	Manet	Inpresionismoa	161x097
Mun1	El grito	1893	Munch	Expresionismoa	091x073
Pic1	Guernica	1937	Picasso	Kubismoa	349x776
Rem1	Ronda noche	1642	Rembrandt	Barrokoa	363x437
Van1	Girasoles	1889	Van Gogh	Inpresionismoa	095x073
Vel1	Meninas	1656	Velazquez	Barrokoa	318x276
Vel2	Hilanderas	1644	Velazquez	Barrokoa	222x293
Vel3	Lanzas	1634	Velazquez	Barrokoa	307x367
Vin1	Gioconda	1506	Da Vinci	Errenazimentu	077x053
Vin2	Última cena	1497	Da Vinci	Errenazimentu	480x880
War1	Marilyn	1964	Warhol	Pop Art	101x101



Oinarrizko indizea B⁺ zuhaitzarekin (oinarrizko gakoa) (2/2)

- Oinarrizko indizea (OI):
 - Adibidean, *ID* gakoarentzako definitu da
 - Bi-g indizean bezala eraiki daiteke (beste aukerak daude)
- Datu-fitxategia: indexazio-eremuaren arabera **ordenatua** (ID). Beraz, fitxategi bakoitzeko **OI bakarra** defini daiteke. Ez da bateragarria erregistroen ordenazio fisikoa eskatzen duten bestelako oinarrizko antolamenduekin (helbideratze kalkulatua, beste eremu bat(zu)en arabera ordenazioa ...)
- Ordenazio logikoa: hemen ez du zentzurik, erregistroak fisikoki ordenatuak baitaude
- Erregistro baten bilatzea: Bi-g indizeetan bezala
- Bilaketa guztietan maila bakoitzeko adabegi bat bisitatzen da + bilatutako balioa duen erregistroa duen blokea (bakarra, gakoa baita)
- $ID > 'Gri1'$ edo adibidez $'Goy1' \leq ID \leq 'Vel3'$ moduko balioak dituzten erregistroak bilatzea ahalbidetzen du
- Erregistroak txertatzea/ezabatzea: edozein indizeetan bezala, indizearen eguneratzea eta berrantolatzea eskatzen du

**Multzokatze-
indizea**
Margolaria
eremuaren
gainean
(B⁺zuhaitza)

Bloke:

2



9



Multzokatze-indizea B⁺ zuhaitzarekin (gakoa ez den eremua) (2/2)

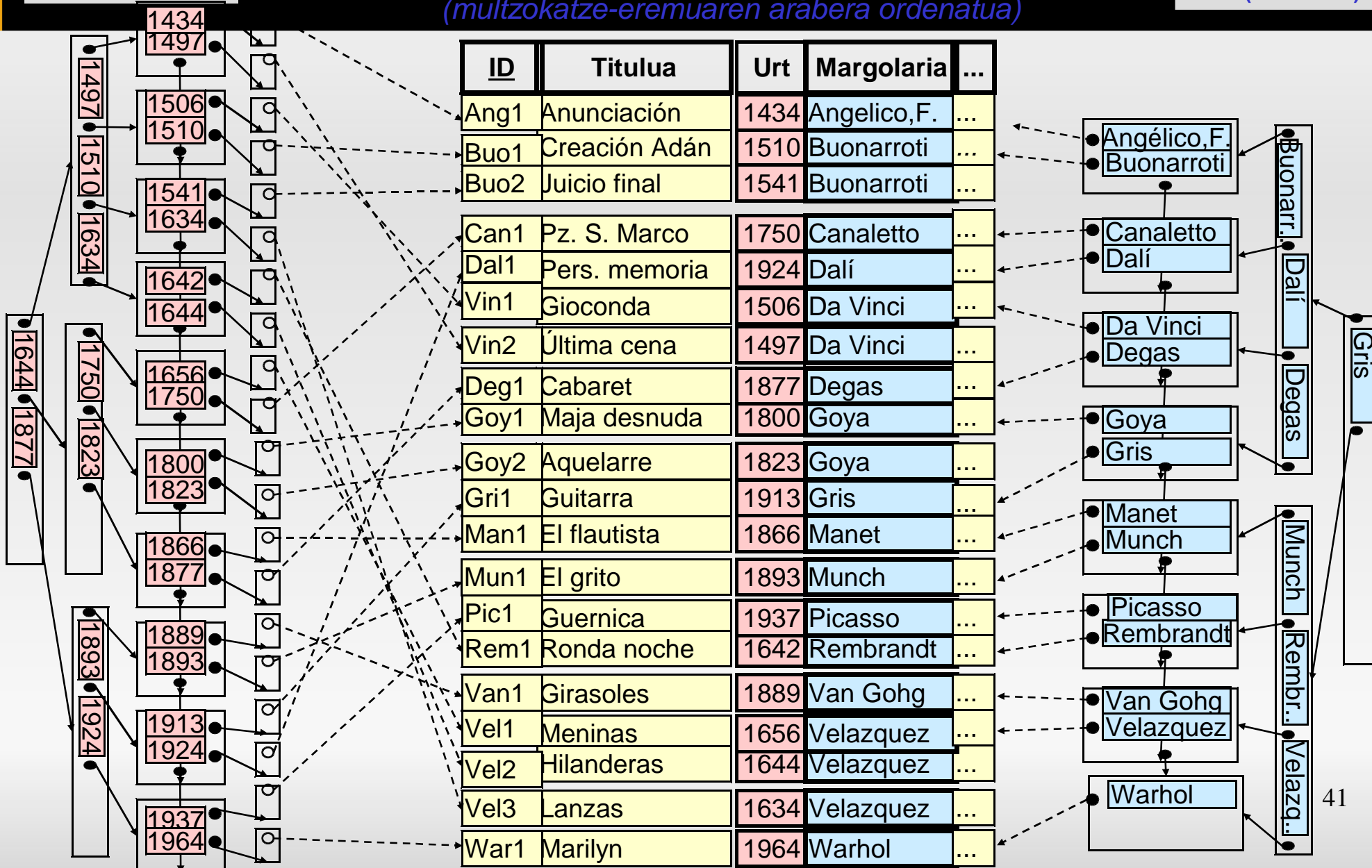
- Multzokatze-indizea (TI):
 - Adibidean, *Margolaria* eremuarentzako definitu da
 - Bi-g indizearen antzera eraiki daiteke, baina hostoetako balio-erakusle bakoitzak, balio hori duen lehen erregistroa duen blokea erakusten du (*Margolarian* balio bera duten hainbat erregistro egon daitezke, ez baita gakoa)
- Datu-fitxategia: indexatze-eremuaren arabera ordenatua (Margolaria). Horrela, *Margolari* beraren erregistro guztiak elkarrekin daude. Beraz, TI **bakarra** defini daiteke fitxategi bakoitzeko. Ez da bateragarria erregistroen ordenazio fisikoa eskatzen duten bestelako oinarritzko antolamenduekin (helbideratze kalkulatu, beste eremu bat(zu)en araberrako ordenazioa ...)
- Ordenazio logikoa: hemen ere ez du zentzurik, erregistroak fisikoki ordenatuak baitaude
- Erregistro baten bilatzea: balio hori duen lehen erregistroa Bi-g indizetan bezala bilatzen da. Datu-fitxategian balio bera duten gainerako erregistroak topatutako lehen erregistroaren segidan doaz.
- Bilaketa guztietan maila bakoitzeko adabegi bat bisitatzen da + bilatutako balioa duten erregistroak (batzuk izan daitezke, ez baita gakoa)
- $\text{Margolaria} > \text{'Gris'}$ edo adibidez, $\text{'Goya'} \leq \text{ID} \leq \text{'Velázquez'}$ moduko baldintzak bilatzea ahalbidetzen du
- Erregistroak txertatzea/ezabatzea: edozein indizeetan bezala, indizearen eguneratzea eta berrantolatzea eskatzen du

Bigarren mailako
INDIZEA gakoa
EZ den Urtea
eremuaren gainean
(B+zuhaitza)

Bateragarritasuna (1/2)

DATU-fitxategia
(multzokatze-eremuaren arabera ordenatua)

Multzokatze-indizea
Margolaria
multzokatze-eremuaren
gainean
(B+zuhaitza)

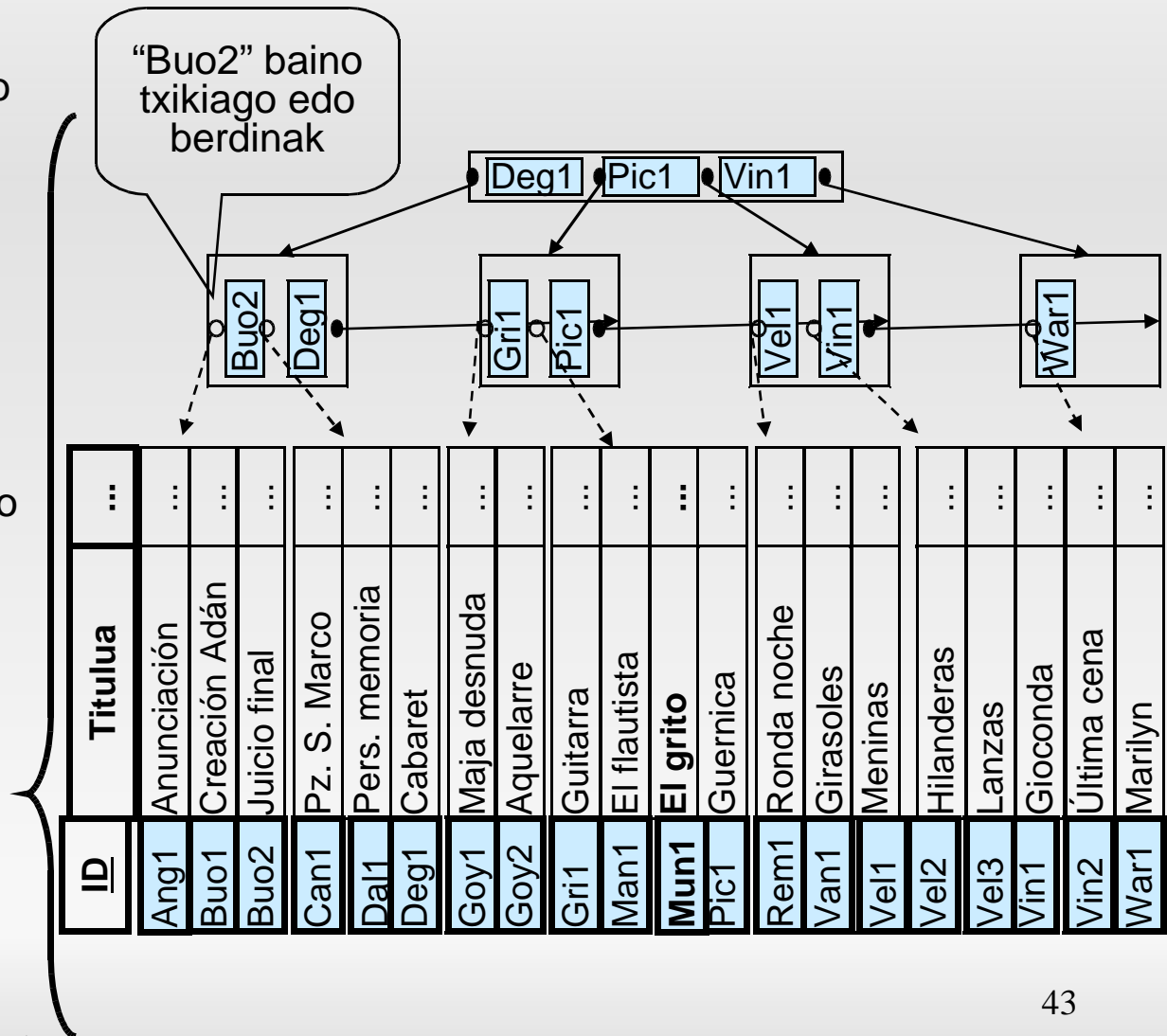


Bateragarritasuna (2/2)

- Aurreko adibidean fitxategian bi indize definitu dira:
 - Multzokatze-indize bat *Margolaria* eremuaren gainean (**TI(Margolaria)**)
 - Bigarren mailako indize bat, gakoa EZ den *Urt* eremuaren gainean (**Bi-eg(Urt)**)
 - Nahi adina bigarren mailako indize defini daiteke
- **Bateraezintasunak zergatik?**
 - Fitxategi bateko erregistroak fisikoki kontzeptu bakarraren arabera soilik ordenatu daitezke
 - Hau da, ezin dira ordenatuak egon *Margolariaren* eta *Tituluaren* arabera aldi berean
- **Beraz, honakoak baiezta ditzakegu:**
 - Bigarren mailako indize bat (Bi-g edo Bi-eg) beste edozein indizerekin bateragarria da, baita beste bigarren mailako indizeekin ere
 - Oinarrizko indizeak, multzokatze-indizeak, fitxategi ordenatuak eta hash-funtzioak erabiltzen duten fitxategiak, bigarren mailako indizeekin bateragarriak dira
 - Ezin dira eduki fitxategi berarentzako bi oinarrizko indize, edo bi multzokatze-indize, edo bi ordenazio desberdin, edo bi helbideratze kalkulatu.

Indize trinko eta ez-trinkoak

- **Indize trinkoa:**
 - Sarrera bat dauka datu-fitxategiko erregistro bakoitzeko
 - Aurreko adibideetan, gakoaren gainean definitutako indizeak trinkoak ziren, eta gako ez ziren eremuen gainean definitutakoak, ez-trinkoak
- **Indize ez-trinkoa:**
 - Trinkoa ez dena, hots, datu-fitxategiko erregistro batzuentzako soilik ditu sarrerak
- **Oinarrizko indize ez-trinko baten** adibidea: indizeko bloketan ez daude datu-fitxategiko erregistro guztiak, batzuk soilik daude. Kasu honetan, erakusten den blokeko balio handiena dago hostoetan, ez balio guztiak. Multzokatze-indizeetarako gauza bera egin daiteke, erregistroak ordenatuta daudela aprobetxatuz.



Indizeak gako konposatuen gainean (1/3)

- **Atributu bat baino gehiagorekin osatutako gakoak:**

- Kotsultetan maiz azaltzen direnak
- Konbinaketarako atzipen-egitura bat sortzea hobe

- **Adibidean:** Margolaria + Urt

- Kotsulta: Velázquez-ek 1644 urtean margotutako margoak
- Ez Margolaria, ezta Urt ere, ez dira gakoak (Margolari baten erregistro asko eta Urt bereko erregistro asko egon daitezke)

- **Estrategiak:**

- 1. Bi-eg(Margolaria) eta Urt indizerik gabe:

Indizea erabili Margolaria="Velázquez" duten erregistroak ekartzeko, eta erregistro horien artean Urt=1644 dutenak bilatu

- 2. Bi-eg(Urt) eta Margolaria indizerik gabe:

Indizea erabili Urt=1644 duten erregistroak ekartzeko, eta erregistro horien artean Margolaria="Velázquez" dutenak bilatu

- 3. Bi-eg(Margolaria) eta Bi-eg(Urt):

Lehen indizearekin Margolaria="Velázquez" duten erregistroak bilatu, eta bigarrenarekin, Urt=1644 dutenak. Gero, erregistroen arteko ebaketa egin (edo horien erakusleena)

- **Arazoa** aldi berean ondorengoa gertatzen bada:

- Velázquez-en margo asko dago
- 1644eko margo asko dago
- Velázquez-ek 1644ean margotutako margo gutxi dago
- **Kasu honetan, hiru estrategia hauek ez dira eraginkorrak**

- **Soluzioa:** atributu anitzen gainean indizea definitzea

Indizeak gako konposatuen gainean (2/3)

DATU-Fitxategia
(multzokatze-eremuaaren arabera ordenatua)

ID	Titulua	Margolaria	Urt	...
Ang1	Anunciación	Angelico,F.	1434	...
Buo1	Creación Adán	Buonarroti	1510	...
Buo2	Juicio final	Buonarroti	1541	...
Can1	Pz. S. Marco	Canaletto	1750	...
Dal1	Pers. memoria	Dalí	1924	...
Vin2	Última cena	Da Vinci	1497	...
Vin1	Gioconda	Da Vinci	1506	...
Deg1	Cabaret	Degas	1877	...
Goy1	Maja desnuda	Goya	1800	...
Goy2	Aquelarre	Goya	1823	...
Gri1	Guitarra	Gris	1913	...
Man1	El flautista	Manet	1866	...
Mun1	El grito	Munch	1893	...
Pic1	Guernica	Picasso	1937	...
Rem1	Ronda noche	Rembrandt	1642	...
Van1	Girasoles	Van Gogh	1889	...
Vel3	Lanzas	Velázquez	1634	...
Vel2	Hilanderas	Velázquez	1644	...
Vel1	Meninas	Velázquez	1656	...
War1	Marilyn	Warhol	1964	...

Angelico,F., 1434
Buonarroti, 1510

Buonarroti, 1541
Canaletto, 1750

Dalí, 1924
Da Vinci, 1497

Da Vinci, 1506
Degas, 1877

Goya, 1800
Goya, 1823

Gris, 1913
Manet, 1866

Munch, 1893
Picasso, 1937

Rembrandt, 1642
Van Gogh, 1889

Velázquez, 1634
Velázquez, 1644

Velázquez, 1656
Warhol, 1964

Buon...
1510

Canal...
1750

Degas,
1877

Goya,
1823

Picasso,
1937

Velaz...
1644

**MULTZOKATZE-
INDIZEA**
(Margolaria, Urt)-en
gainean
(B+ zuhaitza)

Da Vi...
1497

Manet
1866

Van G...
1889

Indizeak gako konposatuen gainean (3/3)

- Aurreko adibidean, indize bat definitu dugu, eremu bat baino gehiagoren gainean
- Multzokatze-indizeez gain, oinarrizko indizeak eta bigarren mailako indizeak defini daitezke
- Gako konposatuko balioak, *lexikografikoki* ordenatzen dira:
 - (“**Dalí**”,1924) < (“**Goya**”,1800) lehen eremuan balio desberdinak dituztenean
 - (“Da Vinci”, **1497**) < (“Da Vinci”, **1506**) lehen eremuan balio berdinak dituztenean
- Atributu anitzekin lan egiteko gainerako egiturak:
 - Hashing zatitua
 - Sareta-fitxategiak (*Grid*)

Ondoren aztertuko ditugu

Hashing zatitua

- Kanpo-hashing estatikoaren hedapena (dispertsioa)
- Berdintasun-alderaketak soilik onartzen ditu
- n atributuekin osatutako atributu konposatuaren gainean definitutako *hash* funtzioa:
 - n helbide sortzen ditu
 - Kaxoiaren helbidea, sortutako n helbideen kateamendua da
- Adibidea:
 - Margolaria="Munch". Lortzen den helbidea = 100 (3 bit)
 - Urt=1893. Lortzen den helbidea= 10101 (5 bit)
 - Kaxoiaren helbidea: 100 10101 (kateatuak)
- Urt= 1893 dutenak soilik bilatzen baditugu, ondorengo kaxoiak aztertu behar dira: **000**10101, **001**10101, ... , **111**10101
- **Abantailak:**
 - Era errazean, edozein atributu kopururekin erabil daiteke
 - Pisu handieneko bitak, gehien atzitzen diren atributuei egokitu diezazkiokegu
 - Ez da beharrezkoa atzipen-egitura bat atributu bakoitzarentzako
- **Desabantaila:**
 - Atributuko konposatuko atributuekin ezin dira berdintasuna modukoak ez diren (<, > ...) konparaketak egin (Adibidea: Urt>1893)

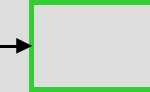
Sareta-fitxategiak (*Grid*) (1/2)

SZ-ren eskala lineala

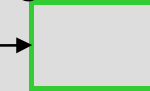
0	1, 2
1	3, 4
2	5
3	6, 7
4	8
5	9, 10

	0	1	2	3	4	5
5						
4						
3						
2						
1						
0						

Kaxoi-gordailua



Kaxoi-gordailua



Langilea fitxategia

Adina-ren eskala lineala

0	1	2	3	4	5
≤20	21-25	26-30	31-40	41-50	>50

Sareta taula

- Bilaketa gako bakoitzarentzako eskala lineal bat eraikitzen da
- Eskala lineala: gako baten balioen banaketa uniformea du
- Sareta taula: Gelaxka bakoitzak kaxoi-helbide bat dauka. Kaxoi horretan, balio horiei dagozkien erregistroak gordetzen dira.
- SZ=4 eta Adina=59 duten erregistroak topatzeko, (1,5) gelaxka begiratzen da. Bertan dagoen helbideak, aztertu behar dugun kaxoia zein den adierazten digu. Kaxoi horretan, SZ=4 eta Adina=59 duten erregistroak topatuko ditugu

Sareta-fitxategiak (2/2)

- Abantaila: gako batentzako eskala lineal bat definitu bada, “>”, “<” moduko galderak egin daitezke
- n bilaketa-gakoetarako:
 - n edozein zenbaki izan daiteke
 - n eskala lineal
 - Taula: n dimentsiotako sareta
- Sareta-aula erabilita, datu-fitxategia banatu dezakegu definitutako bilaketa gakoak erabilita
- Gainera, dimentsiotako balioak konbinatuaz balioak atzitu ditzakegu
- Gako konposatuen gaineko atzipen denbora txikitu egiten da
- Espazio handiagoa behar dute, sareta-aula gorde behar baita
- Fitxategia dinamikoa bada, berrantolaketak maiz egitea beharrezkoa da

Indize fisikokoak eta indize logikoak

Indize fisikoak

- Orain arte ikusitakoak dira. Diskoan erregistroa tokiz aldatzen bada erakuslea eguneratu egin behar da
- Arazoa: eguneraketa asko egin beharra gerta daiteke. Adibidez, 2. mailako indize asko dauzkan hash fitxategi batean

Indize logikoak

- *Adibidez, Sql Server-eko bigarren mailako indizeak, taularentzako oinarritzko indize bat definitu denean.*
- Sarreraren itxura honakoa da <G, Go>, non:
 - G: 2. mailako indexazio-eremua da
 - Go: oinarritzko gakoaren balioa da (oinarritzko antolamenduarena, adib. hash-funtzioa)
- Indize logiko bat 2. mailako indize berezi bat da. G gakoa duen erregistroa bilatzeko orduan dagokion Go oinarritzko gakoa duen erregistroa bilatzen du (G dagoen erregistroarena). Gero, oinarritzko antolamendua erabili beharko da (adib. hash-funtzioa) Go oinarritzko gakoa duen erregistroa bilatzeko
- Helbideratzea egiteko maila berri bat erabiltzen da
- Helbide fisikoak maiz aldatzen direnerako aproposa

5. gaia: Diseinu fisikoa

- *Helburua*
- *Datu-baseen gordeetzea eta fitxategi-antolamendua*
Elmasri & Navathe 09
- *Fitxategietako indexatze-egiturak edo atzipen-egiturak*
Elmasri & Navathe 09
- *Diseinu fisikorako gomendioak eta diseinua nola egin*
Conolly & Begg 05
- *Kontsulta-optimizazio*
Elmasri & Navathe 09

Gomendioak: Biltegi-egitura

- Erregistro ordenatugabeen fitxategia (*heap*)

- BAI {
- Datuak taulan **kantitate handitan kargatu** behar direnean
 - Taulak toki gutxi hartzen duenean
 - Erregistro bat topatzeko denbora laburra izango da
 - Behin taula atzitutakoan, **tupla guztiak jaso** behar direnean

- EZ {
- Taulatik **tupla konkretu batzuk baino jaso nahi** direnean

Gomendioak: Biltegitratze-egitura

■ *Hashing-a*

- BAI {
- Tuplak lortzeko **hash-eremuko balioarekin berdintasuna** edo bat-etortze zehatza behar denean
- EZ {
- Tuplak lortzeko hash-eremuko balioaren **balio-tarte bat** erabili behar denean (<, > ...)
 - Tuplak lortzeko **hash-eremua ez den atributu bat** erabiltzen denean
 - Tuplak lortzeko hash-eremuaren **zati bat** erabili behar denean

Gomendioak: Biltegi-egitura

- B+ zuhaitzadun atzipen-bidea (indizea)
 - Hash-egitura baino erabilera anitzagoa/zabalagoa. Atzipenak edo tupla lortzeak egin daitezke berdintasuna, balio-tarteak eta gako partzialak erabiliaz
 - Indizea **dinamiko** da: taularekin handitzen da
 - Taulan egindako aldaketekin indizeen errendimendua ez da okertzen eta ordena mantendu egiten da bilatze-eremuan
 - Indizera atzipen konkurrentea egiten denean kudeaketa zaila

Gomendioak atzipen-bideari buruz eta diseinu fisikoa

- Ondorengo gardenkietan, atzipen-bide edo indizeak definitzeko gomendioak ematearekin batera, fitxategi baterako diseinu fisikoa egiten ikasiko dugu
- Diseinu fisikoaren **helburua**:

DBko taula bakoitzerako fitxategi mota eta indize egokienak topatzea, sistemaren errendimendu ona lortzeko asmoz

1. urratsa: **TRANSAKZIOAK ANALIZATZEA**

- Helburua: transakzio garrantzitsuenak aztertzea eta ulertzea
- 80/20 araua: transakzio erabilien % 20ak, datuen atzipenaren % 80a egiten du
- % 20 hori soilik aztertzen da (transakzio asko badago)

2. urratsa: **OINARRIZKO FITXATEGI-ANTOLAMENDUA ERABAKITZEA TAULA BAKOITZEKO**

- Helburua: oinarrizko antolamendu bat (fitxategi mota) eraginkor bat aukeratzea taula bakoitzeko

3. urratsa: **TAULA BAKOITZEKO BIGARREN MAILAKO INDIZE EGOKIAK AUKERATZEA**

- Helburua: sistemaren errendimendua hobe dezaketen bigarren mailako indizeak aukeratzea, beharrezkoak baldin badira, noski⁵⁶

1. urratsa: TRANSAKZIOAK ANALIZATZEA (1)

1. Transakzio bakoitzaren exekuzio-maiztasuna

- Adibidez: 50 exekuzio orduko
- Gailurrak ezagutzea garrantzitsua da. Adibidez, 17:00etatik 19:00etara orduko 100 exekuzio egongo dira
- Datu hauekin 80-20 araua aplikatzen da
- Lehentasuna izango dute, indizea izango dute adibidez, maizen atzitzen diren atributuek

2. Zein taula erabitzen du transakzio bakoitzak?

1) **UPDATE** Jabegoa **SET** Kalea=..., Herria=..., PostaKod=..., Mota=..., Gela-kop=..., Errenta=... **WHERE** Szbkia=...
20 alditan egunean

2) **SELECT** L.Kargua, **count**(*)
FROM Delegaritza as D **join** Langilea as L **on** D.Dzbkia = L.Delegaritza
WHERE D.Herria="Iruña"
ORDER BY L.Kargua
500 alditan egunean

3) **SELECT** ...
FROM Jabegoa as J
WHERE J.Mota = ...
50 alditan egunean

Taula	Transakzioa	Maiztasuna <i>Eguneko aldi kop.</i>
Jabegoa	1	20
	3	50
Delegaritza	2	500
Langilea	2	500
...

1. urratsa: TRANSAKZIOAK ANALIZATZEA (2)

3. tauletako atributuetara atzipena

- Atzipen-egiturak (indizeak) izateko hautagaiak:
 - SELECT, UPDATE edo DELETE-ko WHERE-ko atributuak
 - JOIN-ean agertzen diren atributuak
 - Ordenazioa eska dezaketen eragiketetan agertzen diren atributuak: ORDER BY, GROUP BY, UNION, DISTINCT, ...
- Atzipen-egituretan ekidin behar diren atributuak:
 - UPDATE eragiketetan SET aginduan agertzen direnak (aldatzen direnak)

Taula	Transakzioa	Maiztasuna <i>Eguneko aldi kop.</i>	Atributuak
Jabegoa	1	20	where: Szbkia
	3	50	where: Mota
Delegaritzza	2	500	where: Herria join: Dzbkia
Langilea	2	500	join: Delegaritzza order by: Kargua
...

2. urratsa: OINARRIZKO FITXATEGI-ANTOLAMENDUA ERABAKITZEA (1)

- **Adibidea:** *langileak* ordena alfabetikoan lortu nahi baditugu, orduan, *langilea* taula **izena** atributuaren arabera ordenatuta egotea komeni zaigu. Honen ordezt, bi balioen arteko **soldatak** lortu nahi badira, ez du zentzurik taula **izenaren** arabera ordenatzea.
- DBKS batzuek indizeak soilik onartzen dituzte
- Antolamenduen errepasoa:
- **Ordenatu gabeak:**
 - Taulak errenkada gutxi espero baditu
 - Kontsultek errenkada **guztiak** atzitzen badituzte (ordena zehaztu gabe)
- **Hash:**
 - Atributuen gaineko **berdintza** bat hobetzea garrantzitsua bada (where edo join).
 - EZ da erabilgarria atributuak ordenan atzitzeko edo (\leq , \geq , $<$, $>$, between, ...) eragiketetarako
 - Taula oso maiz handitzen edo txikitzen bada, *hedapen dinamikoa* erabiltzea komeni da

2. urratsa: OINARRIZKO FITXATEGI-ANTOLAMENDUA ERABAKITZEA (2)

- **Oinarrizko indizea (OI) edo Multzokatze-indizea (TI):**
 - Bi aukera:
 1. Ordenatu gabeko taula (ordenatugabea/hash) eta beharrezko **BI** (bigarren-mailako) indizeak sortzea
 2. **OI** edo **TI**-rekin ordenatutako taula eta beharrezko **BI**-ak sortzea
 - Zein atributu(ak) aukeratu **OI** edo **TI**rako? (2. aukera):
 - **JOIN**etan erabilienak (eragiketa eraginkorragoa egingo du)
 - Tuplak **ordenan korritzeko** erabilienak (adib.: GROUP BY, ORDER BY...)

3. urratsa: BIGARREN MAILAKO INDIZEAK

Noiz BAI sortu

- Maiz atzitzen den **gako arrotz** bakoitzeko → **join**-etan parte-hartzen dute eta BI batek eraginkorragoa egingo luke. DBKS batzuek BI bat definitzen dute automatikoki gako arrotz bakoitzeko
- WHERE, JOIN, edo ordenazioa eskatzen duten gainerako eragiketetan (ORDER BY, GROUP BY, UNION, DISTINCT, ...) agertzen diren atributuetarako → BI indizea korrituaz edo beraien hostoen **ordenazio logikoa** erabiliz, azken **emaitzak** edo tarteko emaitzak lor litezke .
- Indize bakarreko plana:** kontsulta **indize bakarra korrituta** lor daitekeenean. Adibidea:

```
SELECT Delegaritza, AVG(Soldata)
FROM Langilea
GROUP BY Delegaritza
```

Adibidez **BI**(Soldata, Delegaritza) edo
BI(Delegaritza, Soldata)

- WHERE-an OR edukiz gero, **guztietarako** indizeak edukitzea SOILIK da eraginkorra:

```
SELECT *
FROM Jabegoa
WHERE mota="pisua" OR errenta>600 OR gela-kop>2
```

BI(mota) eta **BI(errenta)** soilik badaude, Jabegoa korritu behar da gela-kop>2ko tuplak lortzeko. Beraz, BI horiek ez lukete kontsulta azkartuko. Bai balioko lukete, OR ordeztu AND eragiketak bagenitu

BI(mota, errenta, gela-kop) edukita, bai OR baita AND eragiketek ere lor dezakete onura

3. urratsa: BIGARREN MAILAKO INDIZEAK

Noiz EZ sortu

- Taula oso txikia bada → Eraginkorragoa izan daiteke taula osoa memorian kargatzea
- Atributuak (UPDATE) edo taula (INSERT/DELETE) oso maiz eguneratzen badira → Indizeak ere eguneratzea suposatuko luke
- Kontsulta baten emaitzak taularen zati handi bat ematen badu (adib. Tuplen % 70a) → Taularen zati handi bat korritu behar da
- Atributuak karaktere-kate oso luzeak baditu (Adib. Paragrafoak) → Indizearen adabegi bakoitzean balio gutxi sartuko lirateke, eraginkortasuna txikituaz
- Ez hain garrantzitsua:
 - Diskoan espazio falta badago → BI indizeek espazioa hartzen dute
 - Zenbat eta BI gehiago, optimizatzailerak denbora gehiago behar du erabakitzeko
- Kasu guztietan: hoberena **probak egitea** da BI indizeek ematen dituzten emaitzak eta BI gabe lortzen direnak konparatuaz
- **Exekuzio-plana** erakusten duten DBKSko erremintak erabiltzea erabilgarria da
- Kontuan izan behar da DBKSak gako atributuetan (primary key eta unique) “unibokotasuna” bermatzeko indizeak sortzen dituela

Diseinu fisikoaren adibidez

(20 transakziotatik gehien erabiltzen den % 20a hartuko dugu)

Langileak 5.000 errenkada ditu 500 bloketan eta

Jabegoak 50.000 errenkada 500 bloketan.

1. Egunean 200 alditan:

SELECT L.*

FROM Langilea AS L **JOIN** Langilea AS G

ON L.Gainbegiratzalea=G.Lzbkia

WHERE G.Izena=@izen AND G.Abizena=@abizen
AND G.Delegaritza=@dele

2. Egunean 300 alditan

SELECT L.*

FROM Langilea AS L

ORDER BY L.Abizena, L.Izena

3. Egunean 1000 alditan

SELECT L.*

FROM Jabegoa AS J **JOIN** Langilea AS L

ON J.Langilea=L.Lzbkia

WHERE L.Izena=@izen AND
L.Abizena=@abizen
AND L.Delegaritza=@dele

4. Egunean 5000 alditan

SELECT J.*

FROM Jabegoa AS J

WHERE J.Herria=@herri AND
J.Errenta<=@errenta_max

Taula	Trans.	Maizt.	Atributuak	Antolak. eta indizeak.
Langilea	1	200	j: Gainbegiratzalea j:Lzbkia w: Izena, Abizena, Delegaritza	OI(Lzbkia) BI-eg(Gainbegiratzalea)
	2	300	o: Abizena, Izena	BI-eg(Abizena, Izena, Delegaritza)
	3	1000	j: Lzbkia w: Izena, Abizena, Delegaritza	
Jabegoa	3	1000	j: Langilea	MI(Herria, Errenta)
	4	5000	w: Herria, Errenta	BI-eg(Langilea)

D(Lzbkia) ere
posiblea da

Sistema erlazioaletarako ohiko gomendioak

- Fitxategiaren antolamenduaren eta indizeen aukeraketa (errepasoa)
 - Aukeratu atributu bat indizea izateko, i) edo erregistroak ordenan lortzeko erabiltzen bada, edo ii) join-etan maiz erabiltzen bada fitxategiaren ordenaziorako atributu moduan. Atributu honentzako edo oinarrizko edo multzokatze-indizea aukeratu.
 - Aukeratu ordenaziokoa ez den atributu bat hautapenerako baldintzetan edo joinetan maiz agertzen bada. Definitu bigarren mailako indizea.
 - Fitxategia maiz eguneratzen bada, minimizatu fitxategi horretarako definitutako indize kopurua.
 - Atributu bat hautaketak egiteko edo join-etarako maiz erabiltzen bada baina ez erregistroak ordenan lortzeko, orduan hash-antolamendua erabil daiteke.

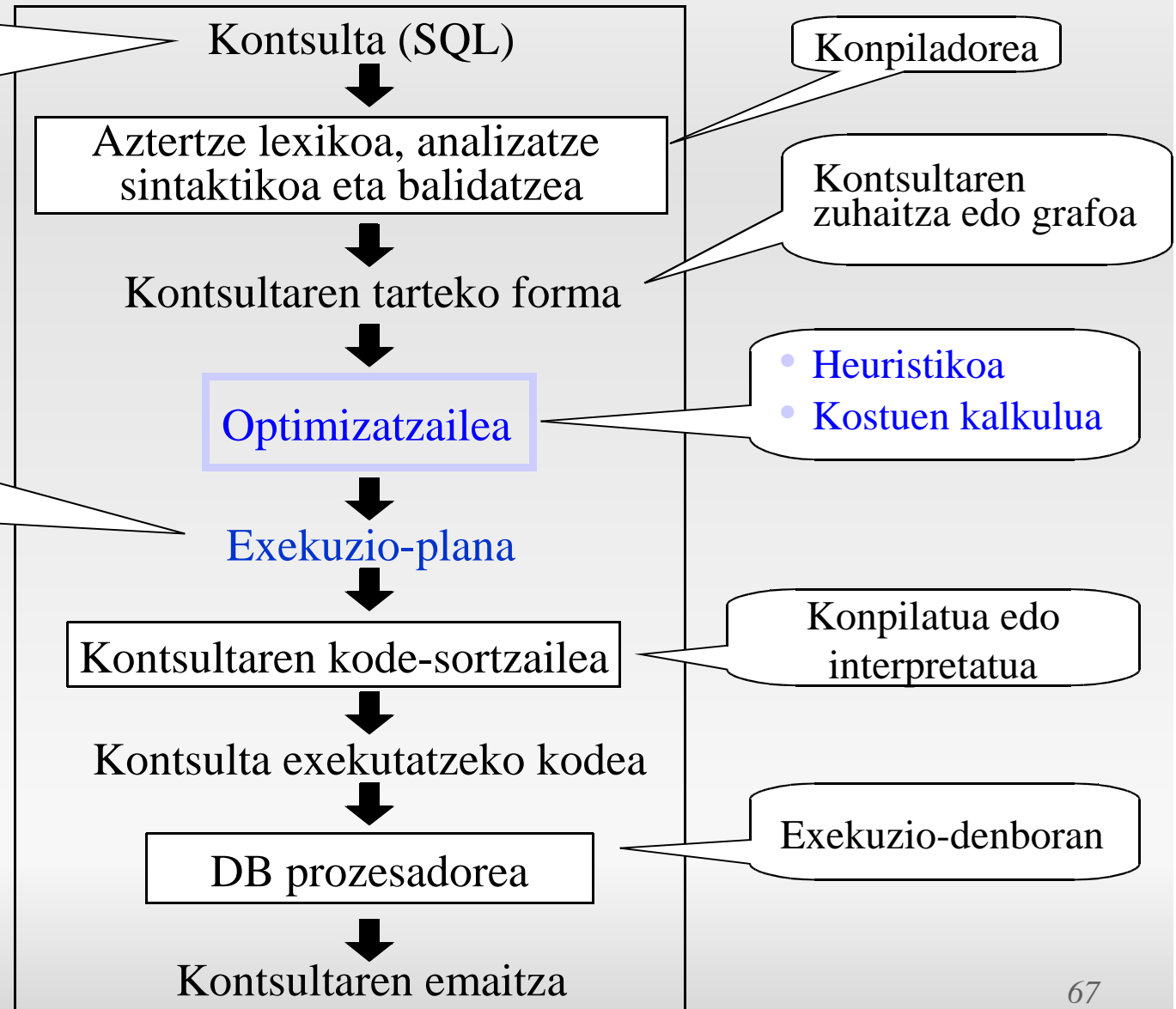
5. gaia: Diseinu fisikoa

- *Helburua*
- *Datu-baseen gordetzea eta fitxategi-antolamendua*
Elmasri & Navathe 09
- *Fitxategietako indize-egiturak edo atzipen-egiturak*
Elmasri & Navathe 09
- *Diseinu fisikorako gomendioak eta diseinua nola egin*
Conolly & Begg 05
- *Kontsulta-optimizazio*
Elmasri & Navathe 09

Goi-mailako kontsulta bat prozesatzeko egin ohi diren urratsak

Zein emaitza espero dugun zehazten da baina **ez nola** lortu behar den

Ez estrategia optimoa, baizik eta **nahikoa** eraginkorra den **estrategia bat**



Optimizazio heuristikoa

- Kontsulta baten **exekuzioa hobetze**arren, bere **barne-errepresentazioa aldatu** egiten da erregela heuristikoak erabiliz
 - **Exekuzio-plan** bat sortzen du, fitxategietarako sortutako atzipen-bide edo indizeetan oinarrituta, eragiketa multzoak exekutatzeko

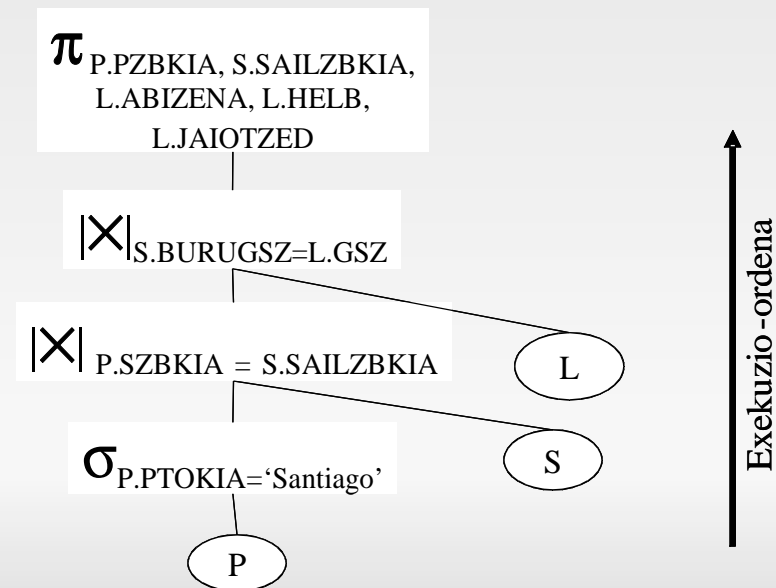
Optimizazio heuristikoaren algoritmoa

- Helburua: bitarteko taulen tamaina txikiagoa egitea
 - **Oinarrizko erregela:**
 - Lehenik σ exekutatu eta gero π (tarteko emaitzak txikitu)
 - Bukaeran $(|\times|, *)$ eta eragiketa bitarrak exekutatu
 - Eragiketa murriztaileenak lehenengo exekutatzen dira eta $(|\times|, *)$ azkenak. Honetarako zuhaitzeko adabegiak berrantolatzen dira.

Kontsulta-zuhaitza

- Aljebra erlazionaleko espresio bat adierazten du
- Kontsulta bera zuhaitz desberdinen bidez adieraz daiteke

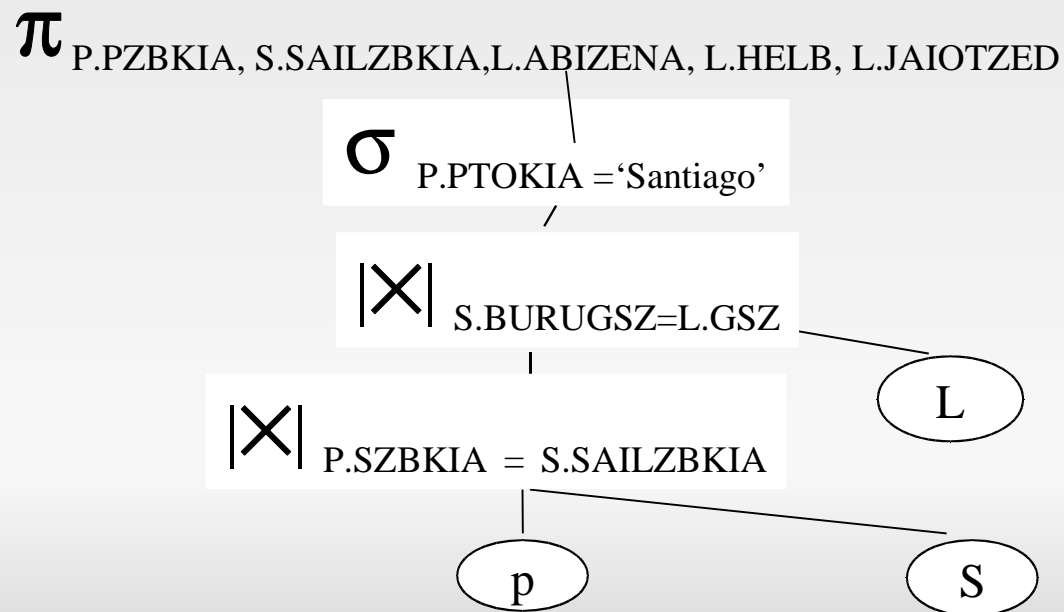
```
SELECT PZBKIA, SAILZBKIA, ABIZENA, HELB, JAIOTZED
FROM (SAILA AS S INNER JOIN PROIEKTUA AS P
      ON P.SZBKIA=S.SAILZBKIA) INNER JOIN LANGILEA AS L
      ON S.BURUGSZ=L.GSZ
WHERE PTOKIA='Santiago';
```



Kontsulta-zuhaitz kanonikoa

- Analizatzaile sintaktikoak sortzen du modu estandarrean
- Inolako optimizaziorik gabe
- Lehenik $|\times|$, gero σ eta azkenik π

```
SELECT PZBKIA, SAILZBKIA, ABIZENA, HELB, JAIOTZED
FROM   (SAILA AS S INNER JOIN PROIEKTUA AS P
        ON P.SZBKIA=S.SAILZBKIA) INNER JOIN LANGILEA AS L
        ON S.BURUGSZ=L.GSZ
WHERE PTOKIA='Santiago';
```



Kontsulta bat eraldatzearen adibidea (1/5)

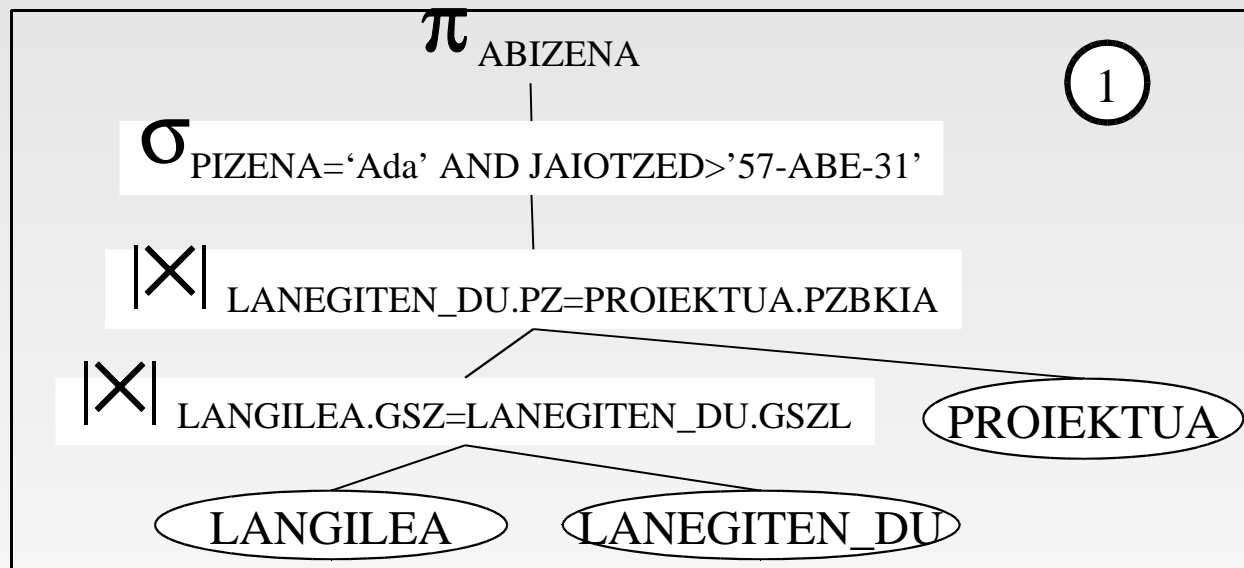
- Hasierako zuhaitza exekuzio eraginkorra duen batean bilakatu
 - **Baliokidetasun-erregelatan** (b.e.) oinarritzen da (algebra erlazionaleko espresioen artean)
 - **Erregela heuristikoak**: baliokidetasun-erregelen aplikazioa gidatzen dute
- Adibidea

1957. urtearen geroztik jaio eta “Ada” izeneko proiektuan lan egiten duten bezeroen abizena lortu

Kontsulta bat eraldatzearen adibidea (2/5)

1957. urtearen geroztik jaio
eta “Ada” izeneko proiektuan lan
egiten duten bezeroen abizena lortu

```
SELECT ABIZENA
FROM (LANGILEA INNER JOIN LANEGITEN_DU ON LANGILEA.GSZ=LANEGITEN_DU.GSZL)
     INNER JOIN PROIEKTUA ON LANEGITEN_DU.PZ=PROIEKTUA.PZBKIA)
WHERE PIZENA='Ada' AND JAIOTZED>'57-ABE-31'
```

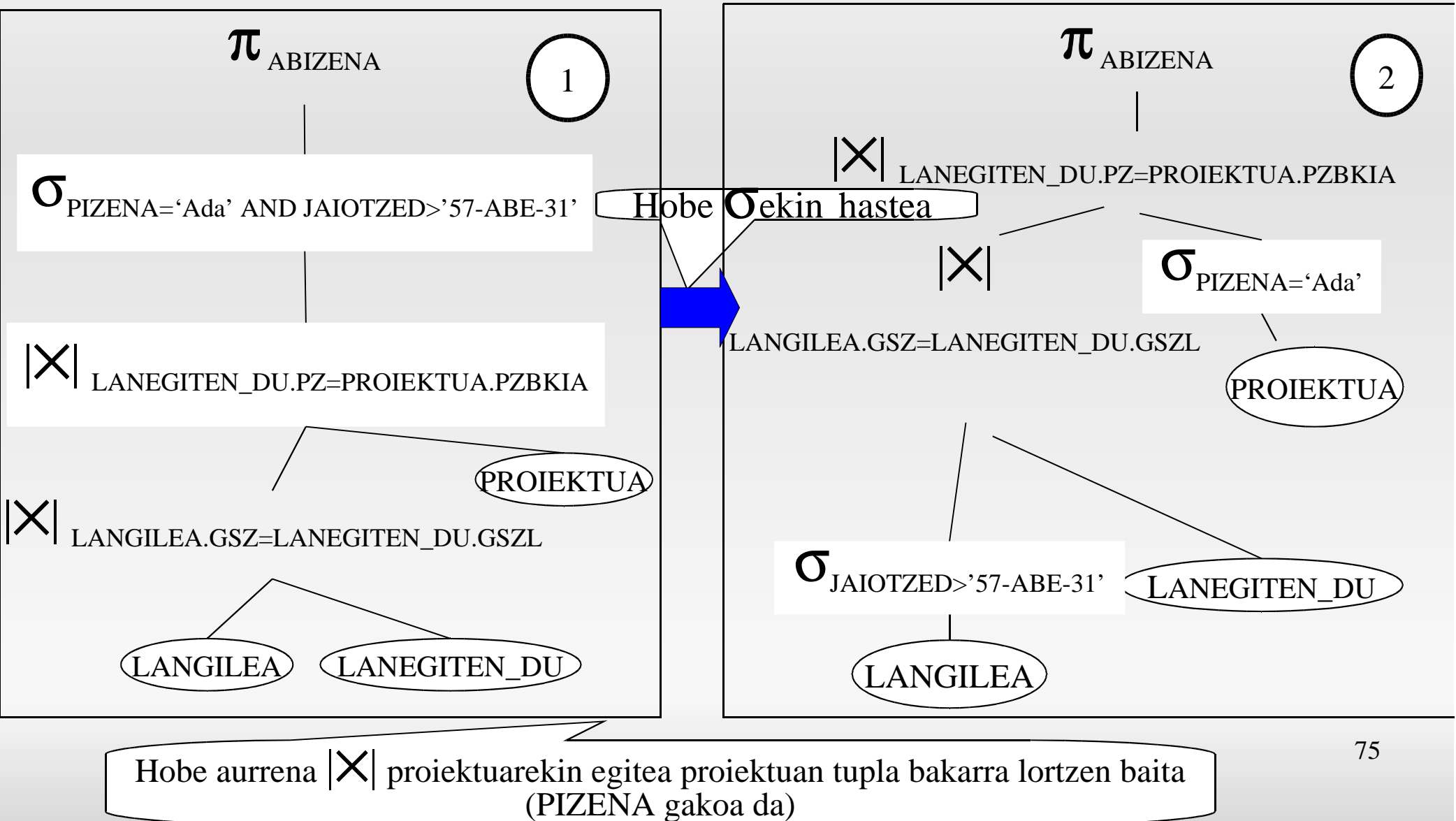


\bowtie fitxategi handiegiak sortzen dituzte.

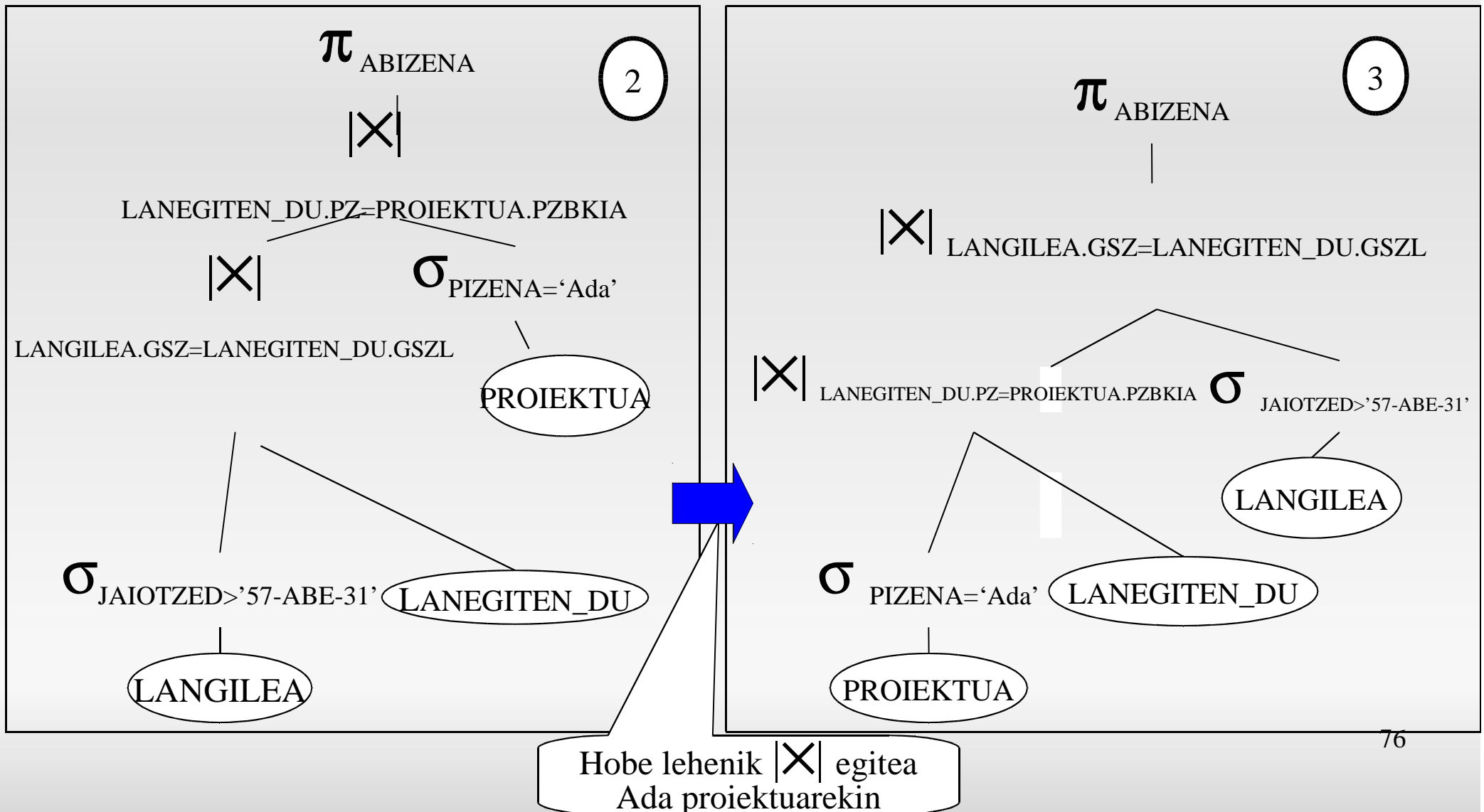
Proiektu bakarria behar da (Ada izeneko) 57. urtearen geroztik jaio direnak bakarrik elkartu

Hobe σ ekin hastea

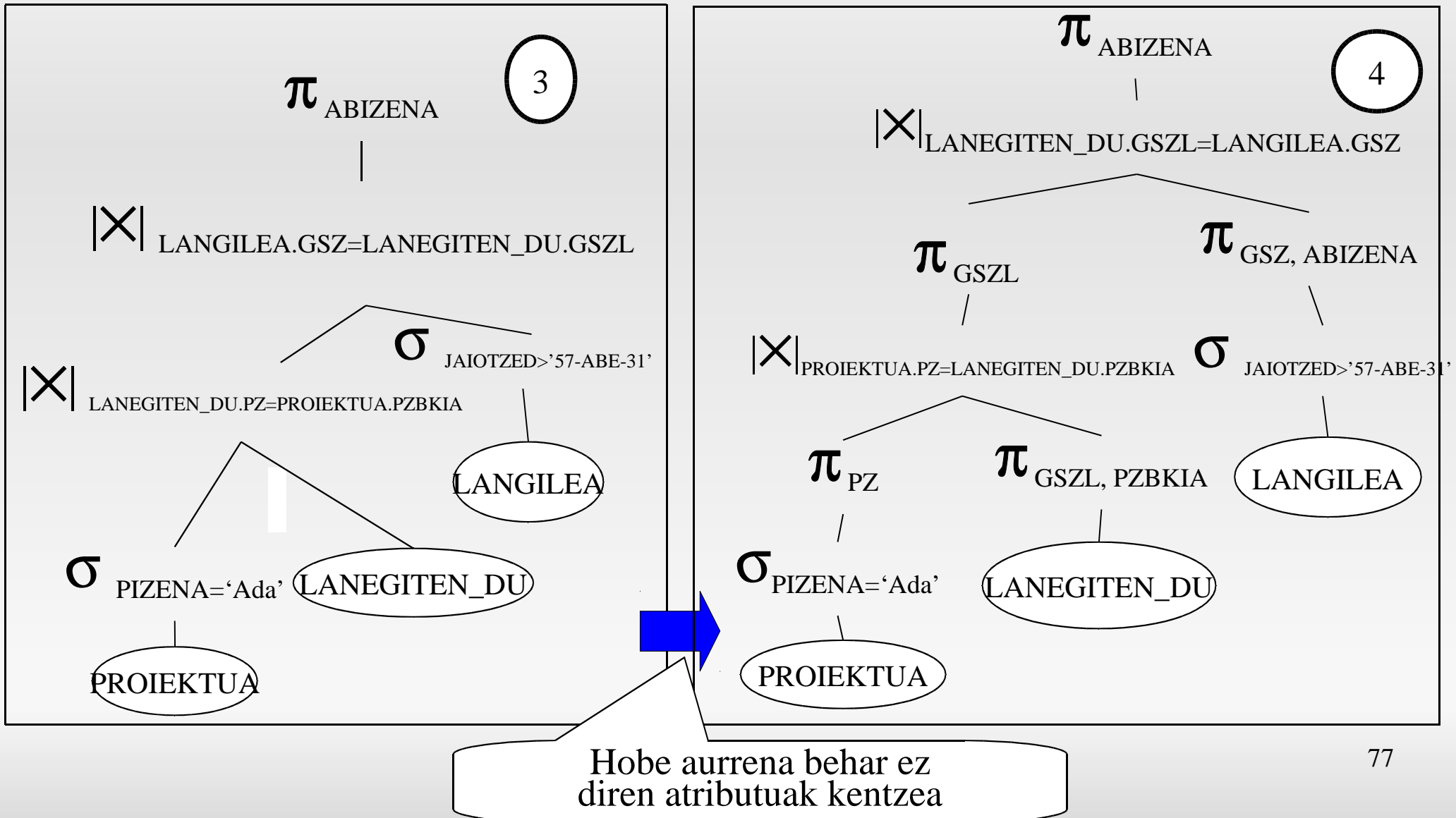
Kontsulta bat eraldatzearen adibidea (3/5)



Kontsulta bat eraldatzearen adibidea (4/5)



Kontsulta bat eraldatzearen adibidea (5/5)



Ariketak

Diseinu fisikoko ariketetan erabiliko dugun notazioa

OG:	Erregistro OrdenatuGabeen fitxategia
O (eremua):	Erregistro Ordenatuen fitxategia(ordenazio-eremua)
D (eremua):	Zuzeneko fitxategia edo hash (Dispertsio- edo kalkulu-eremua)
OI (eremua):	Oinarrizko indizea (indexazio eremua)
MI (eremua):	Multzokatze-indizea (multzokatze-eremua)
BI-eg (eremua):	Bigarren mailako Indizea (Gakoa Ez den eremu baten gainean)
BI-g (eremua):	Bigarren mailako Indizea (Gakoa den eremu baten gainean)

Ariketa: Ospitalea DBan diseinu fisikoa

Ondorengo taulak ditugu:

Gaixoa (GSZ, Izena, Adina, *Sendagile-NAN*) - 5000 errenkada

Sendagaia (Kodea, S-Izena, Dosia) - 10.000 errenkada

Gaixotasuna (G-Izena, Deskribapena) - 500 errenkada

Sendagilea (Sendagile-NAN, Sendagile-izena, Espezialitatea) - 50 errenkada

Tratamendua (GSZ, G-Izena, Kodea, Iraupena) - 100.000 errenkada

Proposa ezazu DB honetarako errepresentazio fisiko bat ondorengo eragiketak modu eraginkorrean egin ahal izateko:

1) Egunean 500 alditan

```
SELECT G.*  
FROM Gaixoa AS G JOIN Sendagilea as S ON G.Sendagile-NAN=S.Sendagile-NAN  
WHERE S.Izena=@Sendizen
```

2) Egunean 2.000 alditan

```
SELECT T.G-Izena, S.S-Izena  
FROM (Gaixoa AS G JOIN Tratamendua AS T ON G.GSZ=T.GSZ) JOIN Sendagaia AS S ON T.Kodea=S.Kodea  
WHERE G.Izena=@gaixolzen  
ORDER BY T.G-Izena, S.S-Izena
```

3) Egunean 1.000 alditan

```
SELECT S.Espezialitatea, S.Sendagile-izena  
FROM (Tratamendua AS T JOIN Gaixoa AS G ON T.GSZ=G.GSZ) JOIN Sendagilea AS S  
      ON G.Sendagile-NAN=S.Sendagile-NAN  
WHERE T.G-Izena=@gaixotasunIzena  
ORDER BY S.Espezialitatea
```

Ariketa: Bidaia-agentzia (1)

Ondorengo taulak ditugu :

Bidai-agentzia (B_A_zbkia, B_A_izena, B_A_helb, Zuzendaria) - *1000 errenkada*

Bezeroa (B_NAN, Izena, Ab1, Ab2, Helb, Tlfnoa) - *20.000 errenkada*

Hotela (H_Izena, Hiria, Kategoria, Gela_kop) - *1.000 errenkada*

Erreserba (H_Izena, Hiria, B_A_zbkia, B_NAN, Data) - *100.000 errenkada*

Ondorengo eragiketak egiteko, zein da fitxategi antolamendurik egokiena?

- 1) Hotel baten izena eta hiri baten izena emanda, hotel horretarako erreserbak egin dituzten bezeroen izenak lortu (egunean 10.000 alditan)

```
SELECT B.Ab1, B. Ab2, B.Izena  
FROM Erreserba AS E JOIN Bezeroa as B ON E.B_NAN=B.B_NAN  
WHERE E.H_Izena=@hotela AND E.Hiria=@hiri
```

- 2) Hasiera eta amaiera data pare baten artean erreserbaren bat egin zuten bezeroen izenak lortu (egunean 5.000 alditan)

```
SELECT B.Ab1, B. Ab2, B.Izena  
FROM Erreserba AS E JOIN Bezeroa as B ON E.B_NAN=B.B_NAN  
WHERE E.Data>=@has AND E.Data<=@buk
```

Ariketa: Bidaia-agentzia (2)

Ondorengo taulak ditugu :

Bidai-agentzia (B_A_zbkia, B_A_izena, B_A_helb, Zuzendaria) - 1000 errenkada

Bezeroa (B_NAN, Izena, Ab1, Ab2, Helb, Tlfnoa) - 20.000 errenkada

Hotela (H_Izena, Hiria, Kategoria, Gela_kop) - 1.000 errenkada

Erreserba (H_Izena, Hiria, B_A_zbkia, B_NAN, Data) - 100.000 errenkada

Ondorengo eragiketak egiteko, zein da fitxategi antolamendurik egokiena?

3) Hotelen izenak alfabetikoki ordenatuta eta errepikapenik gabe (Egunean 100 alditan)

```
SELECT DISTINCT H.H_Izena
FROM Hotela AS H
ORDER BY H.H_Izena
```

4) Telefono bat emanda, telefono hori duten bezeroen izena lortu. Bezero bakoitzeko, erreserbaren bat egin dioten bidai agentzien izena eta helbidea lortu (Egunean 500 alditan)

```
SELECT B.Ab1, B.Ab2, B.Izena, BA.B_A_izena, BA.B_A_Helb
FROM (Bezeroa AS B JOIN Erreserba as E ON B.B_NAN=E.B_NAN)
JOIN Bidai-agentzia AS BA ON E.B_A_zbkia=BA.B_A_zbkia
WHERE B.Tlfnoa=@tfnua
```


Ariketa: Kirol probak (1)

 Gako hautagaiak

Ondorengo taulak ditugu :

Kirolaria (K_zbkia, Izena, Adina, Herrialdea) - 8.000 errenkada

Estadioa (E_kodea, E_izena, Hiria, Edukiera) - 10 errenkada

Proba (P_kodea, Kategoria, Izena, Mota) - 200 errenkada

Esleipena: K_zbkia, P_kodea, E_kodea (24.000 errenk.)

Ondorengo eragiketak egiteko, zein da fitxategi antolamendurik egokiena?


1) Egunean 100 alditan

```
SELECT K.Herrialdea, K.Izena  
FROM Kirolaria AS K  
ORDER BY K.Herrialdea, K.Izena
```

2) Egunean 200 alditan

```
SELECT E.E_izena, E.Hiria, P.Izena, P.Kategoria, P.Mota  
FROM (Estadioa AS E JOIN Esleipena AS ES ON  
      E.E_kodea=ES.E_kodea) JOIN Proba AS P ON ES.P_kodea=P.P_kodea
```

Ariketa: Kirol probak (1)

 Gako hautagaiak

Ondorengo taulak ditugu :

Kirolaria (K_zbkia, Izena, Adina, Herrialdea) - 8.000 errenkada

Estadioa (E_kodea, E_izena, Hiria, Edukiera) - 10 errenkada

Proba (P_kodea, Kategoria, Izena, Mota) - 200 errenkada

Esleipena: K_zbkia, P_kodea, E_kodea (24.000 errenk.)

Ondorengo eragiketak egiteko, zein da fitxategi antolamendurik egokiena?

3) Egunean 1000 alditan

```
SELECT P.*
```

```
FROM Esleipena AS E JOIN Proba AS P ON E.P_kodea=P.P_kodea
```

```
WHERE E.K_zbkia=@kirolariKod
```

4) Egunean 10000 alditan

```
SELECT K.Izena, K.Herrialdea
```

```
FROM (Esleipena AS ES JOIN Estadioa AS E ON ES.E_kodea =E.E_kodea) JOIN Kirolaria AS K  
ON ES.K_zbkia=K.K_zbkia
```

```
WHERE E.E_izena=@estadiolzena AND E.Hiria=@hiri AND ES.P_kodea=@proba
```

```
ORDER BY K.Izena
```

Ariketa:Tximeletak (1)

Espezie (Izena, Ezaugarriak) - 10.000 errenkada

Tximeleta (IzenZientifikoa, Jatorria, Habitat, Biziltzaropena, *Espezielzena*) - 100.000 errenk.

Pertsona (NAN, Izena, Helbidea, Tfnoa, *BildumaKod*, *NagusiaDa*) – 4.500 errenkada

Bilduma (Kodea, Kokapena, Salneurria) - 3.000 errenkada

Ale (IzenZientifikoa, Zbkia, Sorlekua, Tamaina, BildumaKod) – 6.000 errenkada

1) Egunean 1000 alditan

SELECT DISTINCT Esp.Izena, Esp.Ezaugarriak

FROM ((Pertsona AS P **JOIN** Ale AS A **ON** P.BildumaKod=A.BildumaKod) **JOIN** Tximeleta AS T

ON A.IzenZientifikoa=T.IzenZientifikoa) **JOIN** Espezie AS Esp **ON** T.Espezielzena=Esp.Izena

WHERE P.NAN=@bildumaZalea

2) Egunean 2000 alditan

SELECT DISTINCT B.Kodea, B.Salneurria

FROM (Tximeleta AS T **JOIN** Alea AS A **ON** T.IzenZientifikoa=A.IzenZientifikoa)

JOIN Bilduma AS B **ON** A.BildumaKod=B.Kodea

WHERE T.Habitat=@habitat

Ariketa:Tximeletak (1)

Espezie (Izena, Ezaugarriak) - 10.000 errenkada

Tximeleta (IzenZientifikoa, Jatorria, Habitat, BiziItxaropena, *EspezieiIzena*) - 100.000 errenk.

Pertsona (NAN, Izena, Helbidea, Tfnoa, *BildumaKod*, *NagusiaDa*) – 4.500 errenkada

Bilduma (Kodea, Kokapena, Salneurria) - 3.000 errenkada

Ale (IzenZientifikoa, Zbkia, Sorlekua, Tamaina, BildumaKod) – 6.000 errenkada

3) Egunean 800 alditan

```
SELECT P.NAN, P.Izena, P.BildumaKod  
FROM Pertsona AS P  
ORDER BY P.NAN
```

4) Egunean 20000 alditan

```
SELECT A.*  
FROM Bilduma AS B JOIN Ale AS A ON B.Kodea= A.BildumaKod  
WHERE B.Kodea=@bilduma
```

Ariketa: Janaria (1)

Artikulu (ArtIzena, Mota, Prezioa, Tamaina) - 50 errenkada

Osagai (OsaIzena, Prezioa, Hornitzailea) – 500 errenkada

Eskaera (Zbkia, Data, EskaeraOrdua, ErantzunOrdua, Tfnoa) – 100000 errenkada

Esk_Art (Zbkia, ArtIzena, Kantitatea) – 300000 errenkada

Art_Osa (ArtIzena, OsaIzena) - 1.000 errenkada



Gako hautagaiak

...

Ondorengo eragiketak egiteko, zein da fitxategi antolamendurik egokiena?:

1) Egunean 1.000 alditan

SELECT O.Hornitzailea, O. Osaglzena

FROM Osagai AS O

ORDER BY O.Hornitzailea, O. Osaglzena

2) Egunean 300 alditan

SELECT A.Tamaina

FROM (Artikulu AS A **JOIN** Art_Osag AS AO **ON** A.ArtIzena=AO.ArtIzena) **JOIN** Osagai AS O

ON AO.Osaglzena=O.Osaglzena

WHERE O.Prezioa = @prezioa

Ariketa: Janaria (2)

Artikulu (ArtIzena, Mota, Prezioa, Tamaina) - 50 errenkada

Osagai (OsaIzena, Prezioa, Hornitzailea) – 500 errenkada

Eskaera (Zbkia, Data, EskaeraOrdua, ErantzunOrdua, Tfnoa) – 100000 errenkada

Esk_Art (Zbkia, ArtIzena, Kantitatea) – 300000 errenkada

Art_Osa (ArtIzena, OsaIzena) - 1.000 errenkada



Gako hautagaiak

...

Ondorengo eragiketak egiteko, zein da fitxategi antolamendurik egokiena?:

1) Egunean 100 alditan

SELECT A.ArtIzena, A.Prezioa

FROM Artikulu AS A

ORDER BY A.ArtIzena

2) Egunean 5.000 alditan

SELECT E.*, **SUM**(EA.Kantitatea * A.Prezioa) AS EskaeraPrezioa

FROM (Eskaera AS E **NATURAL JOIN** Esk_Art AS EA) **NATURAL JOIN** Artikulu AS A

WHERE E.Zbkia = @eskaera

GROUP BY E.Zbkia