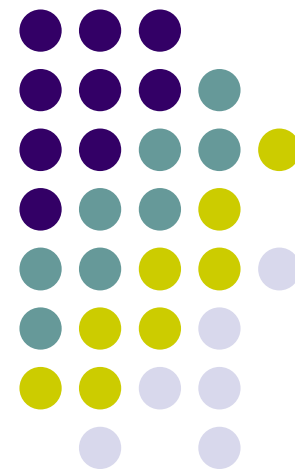
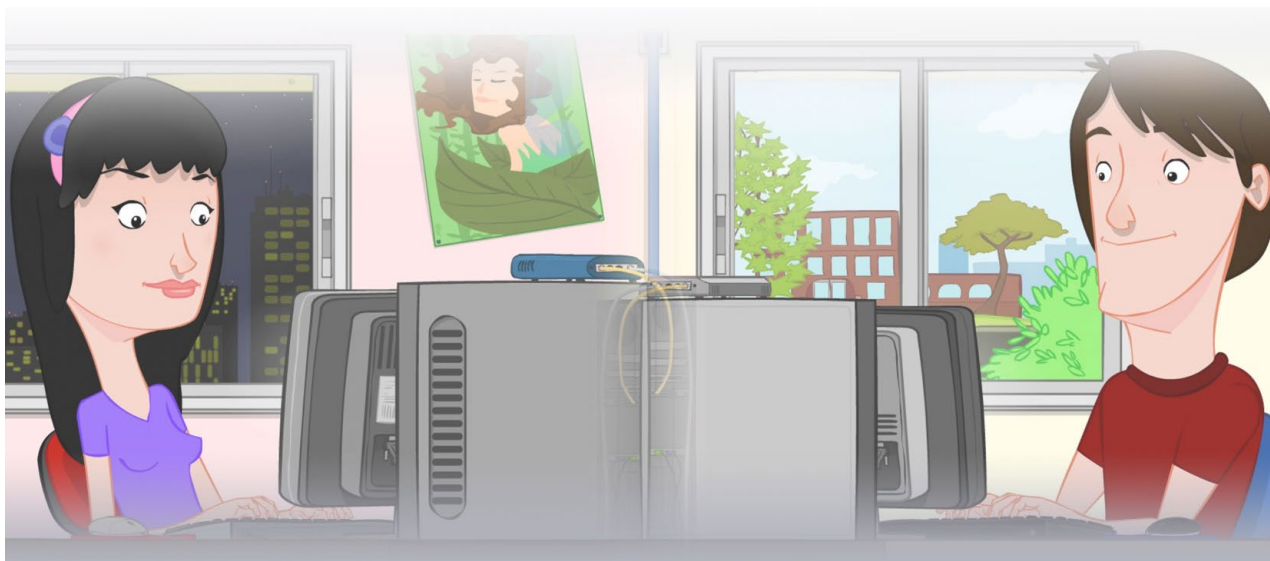


1. Blokea: Sistema Eragileak

Informazio-Sistemen Arkitektura

Telekomunikazio Teknologiaren Ingeniaritzako Gradua (3. Maila)



TELEK:O
UPV/EHU Bilbao



1. Blokea - Edukiak

1. Ordenagailuen Arkitekturako kontzeptuak
2. Sistema Eragileak. Sarrera.
3. Prozesuak
 - Prozesuak eta hariak
 - Prozesuen arteko komunikazioa
 - Komunikazio eta Sinkronizazio Mekanismoak
 - Planifikazioa
4. Sarrera/Irteera
5. Memoriaren Kudeaketa
 - Memoria birtuala
 - Orrikapena eta Segmentazioa
 - Ordezkapeneko algoritmoak
6. Fitxategi Sistema
 - Fitxategiak eta direktorioak
 - Fitxategi Sistemaren antolaketa



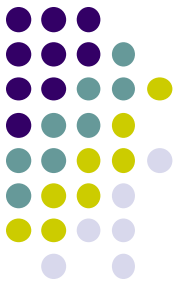
6. Gaia – Edukiak

FITXATEGI-SISTEMA

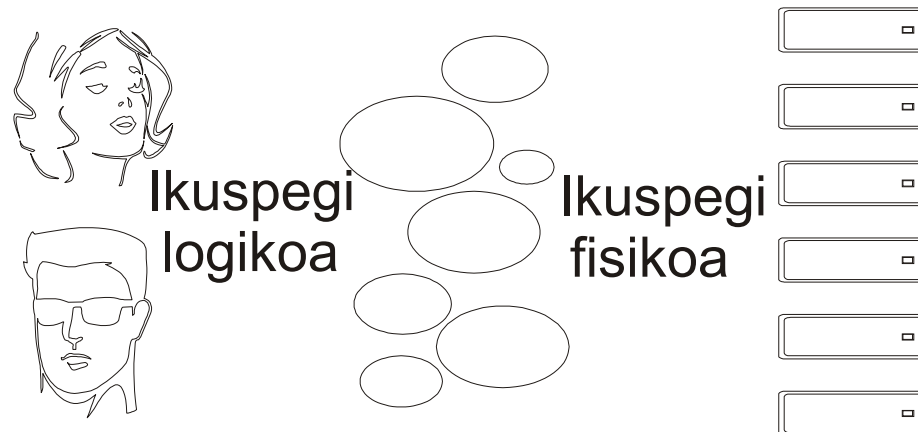
1. Fitxategi-sistema. Erabiltzailearen ikuspuntua.
2. Fitxategiak
3. Direktorioak
4. Fitxategien zerbitzuak
5. Direktorioen zerbitzuak
6. Fitxategi-sistemak
7. Fitxategi-zerbitzaria

6. FITXATEGI-SISTEMA

6.1 ERABILTZAILEAREN IKUSPUNTUA

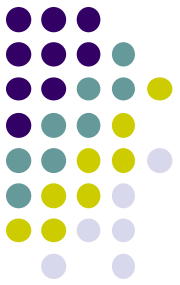


- Ikuspegi logikoa:
 - Fitxategiak
 - Direktorioak
 - Fitxategi-sistemak eta partizioak
 - Gailu bereziak
- Ikuspegi fisikoa:
 - Gailuetan barreiatutako blokeak edo byteak



6. FITXATEGI-SISTEMA

6.1 ERABILTZAILEAREN IKUSPUNTUA



- Gailuak zuzenean atzitzea:
 - Gogaikarria, deserosoa
 - Gailuen zehaztasun fisikoak
 - Helbide fisikoen menpekoa
 - Ez segurua
 - Erabiltzaileak ez du mugarik maila fisikoa atzitzeko gai bada
- Fitxategi-sistema gailuen eta erabiltzaileen arteko software geruza da. Helburuak:
 - Gailuen ikuspegi logikoa eskaini
 - Atzipeneko primitibo erosoak eskaini, zehaztasun fisikoekiko independenteak
 - Babes mekanismoak eskaini

6. FITXATEGI-SISTEMA

6.2 FITXATEGIAK



- Fitxategia: Izen baten azpian informazio elkarlotuta metatzen duen biltegitratze-unitate logiko eta ez-hegakorra.
- Datuak gordetzeko erabiltzen den helbide jarraitu logikoen espazioa
- Fitxategi-motak:
 - Karakterekoa edo ASCII fitxategiak
 - Bitarrak
 - Fitxategi bereziak (karakterekoak/blokekoak)

6. FITXATEGI-SISTEMA

6.2 FITXATEGIAK



- Atributuak:
 - **Izena:** pertsonak batek ulertzeko moduko atributu bakarra.
 - **Fitxategiaren eta erabiltzailearen identifikazio bakarra:** fitxategiaren, jabearen eta taldearen barne-deskriptorea
 - **Fitxategi-mota:** fitxategi-formatu desberdinak antolatzen dituzten sistemetan (zenbaki magikoa).
 - **Fitxategiaren mapa:** gailuaren erakuslea eta blokeen kokapena
 - **Babesa:** atzipen-kontrola eta fitxategien gaineko eragiketen kontrola
 - **Fitxategiaren luzera:** byte-kopurua, gehienezko luzera, etab.
 - **Atzipenei buruzko informazioa:** sortze-unea, azken atzipena, aldaketa, etab.
 - **Kontrolako informazioa:** ezkutuko fitxategia, sistemarena, arrunta, direktorioa.
- UNIX-en i-nodo deritzen egituratan gordetzen dira
 - i-nodo zenbakia fitxategiaren identifikatzaile bakarra izanik

6. FITXATEGI-SISTEMA

6.2 FITXATEGIAK



- Izenak:
 - Fitxategi-sistema bakoitzaren ezaugarria.
 - Izen-mota eta luzera desberdinak, aldeak sistema batetik bestera
 - Luzera: finkoa MS-DOS-en edo aldakorra UNIX-en
 - Luzapena: beharrezkoa ala ez, bat edo gehiago, bana fitxategi-motako, etab.
 - Sentikorra tipografiarekiko ala ez. Adib: MARTIN eta martin.
 - Fitxategi-sistemak barne deskribatzaileak erabiltzen ditu, formatu gutxi batzuk ezagutzen ditu (exekutagarriak, testua...). Adib: UNIX-eko zenbaki magikoa.
 - Direktorioek izen logikoak barne-deskribatzaileekin lotzen dituzte
 - Luzapenak aplikazioetarako esanguratsuak (html, c, cpp, C...).

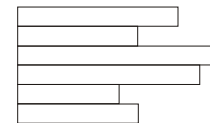
6. FITXATEGI-SISTEMA

6.2 FITXATEGIAK



- Egiturak:

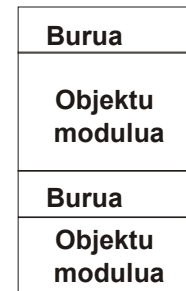
- Bat ere ez. Hitz edo byte-sekuentzia (UNIX)
- Erregistroen egitura sinpleak
 - Lerroak
 - Luzera finkokoak
 - Luzera aldakorrekoak
- Egitura konplexuak
 - Formatudun dokumentuak (HTML, postscript...)
 - Kode-liburutegiak
- Egitura laua eta kontrol-sekuentziak erabiliz egitura konplexuak simula daitezke
- Zeinek erabakitzen du egitura?
 - Barnekoa: Sistema eragileak
 - Kanpokoak: Aplikazioek



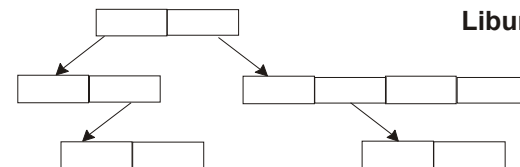
Luzera desberdineko
erregistroak



Byte edo
luzera finkoko
erregistro-sekuentzia



Liburutegia



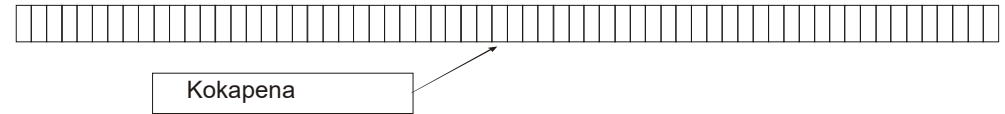
Erregistroen arbolak

6. FITXATEGI-SISTEMA

6.2 FITXATEGIAK



- Egitura logikoa :



- Ikuspegi fisikoa:

- Blokea

- Transferentzia-unitatea
- 2^n sektore
- Fitxategi-sistemak jarritako parametroa

- Zona

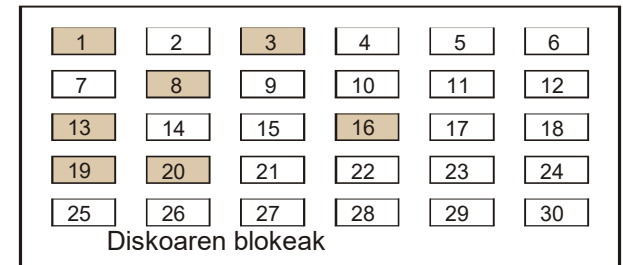
- Esleipen-unitatea
- 2^p bloke
- Fitxategiak sekuentzialagoak izateko

- Egitura fisikoaren deskribapena:

- Erabilitako blokeen antolaketa

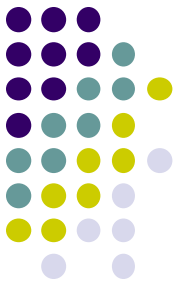
A fitxategia

Blokeak: 13
20
1
8
3
16
19



6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



- Fitxategi izen bat eta bere barne-deskribatzailea unibokoki lotzen duen objektua
- Fitxategi-sistema antolatu eta egituraketari buruzko informazioa ematen du
- Fitxategiei buruzko informazioa duen nodo-bilduma
- Eskema hierarkikoa
 - Fitxategi bat irekitzerakoan SE-ak bere izena bilatu behar du direktorioaren egituran.
- Direktorioen gaineko eragiketak:
 - Sortu (txertatu) eta ezabatu (kanporatu), ireki eta itxi, direktorioa berrizendatu, sarrerak irakurri, muntatu (konbinatu)
- Direktorioen antolaketa hierarkikoak
 - Fitxategien izendatzea errazten du (izen bakarrak)
 - Banaketaren kudeaketa erraztu → antolaketa irizpide logikoen arabera (erabiltzaile bera, aplikazio bera ...)

6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



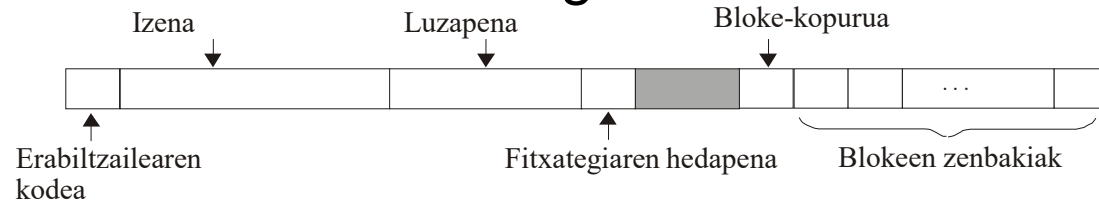
- Egiturak
 - Bai direktorioen bai fitxategien egitura diskoetan gordetzen da.
 - Direktorioak fitxategi moduan inplementatzen dira.
 - Direktorio bateko informazioa: izena, mota, helbidea, luzera maximoa eta unekoa, atzipenen eta aldaketaren denborak, jabea, etab.
- Direktorio egitura oso desberdinak daude. Informazioa egitura horren menpe.
- Bi alternatiba:
 - Fitxategiaren atributuak direktorioko sarreran gorde
 - <izen, identifikatzaile> bikoteak gorde, fitxategiaren datuak beste egitura batean utziz. Hobeto.
 - Fitxategiko atributuetan eginiko aldaketek ez dute eraginik direktorioko sarreretan

6. FITXATEGI-SISTEMA

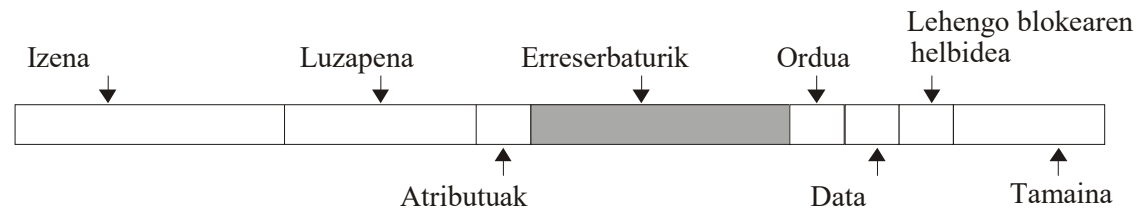
6.3 DIREKTORIOAK



- Direktorioko sarrera bakoitzaren egitura:



CP/M sistemako direktorioa (ZAHARKITUA)



MS-DOS-eko direktorioa



UNIX-eko direktorioa

6. FITXATEGI-SISTEMA

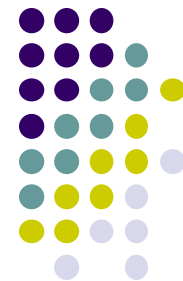
6.3 DIREKTORIOAK



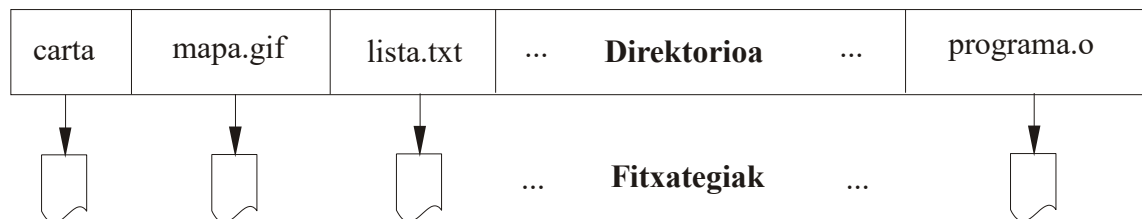
- Antolaketa irizpideak
 - Eraginkortasuna: fitxategiak bizkor bilatu
 - Izendatzea: erabiltzaileentzat erreza eta eroso
 - Bi erabiltzailek izen bera fitxategi desberdinetarako
 - Fitxategi berak izen desberdinak
 - Luzera aldakorreko izenak
 - Taldekatzea: fitxategiak taldeka sailkatze logikoa ezaugarrien arabera (abib: Pascalez idatzitako programak, jokoak, etab)
 - Sinpletasuna: direktorioetako sarrerak ahal bezain sinpleak izan behar dira.

6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



- Maila bakarreko antolaketa
 - Direktorio bakarra fitxategi sarrera guztientzat
 - Erabiltzaile guztiek partekatzen dute direktorio berbera
 - Abantaila: Inplementatzeko erraza eta uler erraza
 - Desabantaila: Izendatzeko, taldekatzeko eta partekatzeko arazoak
 - Konponbidea: Fitxategi kopurua mugatua

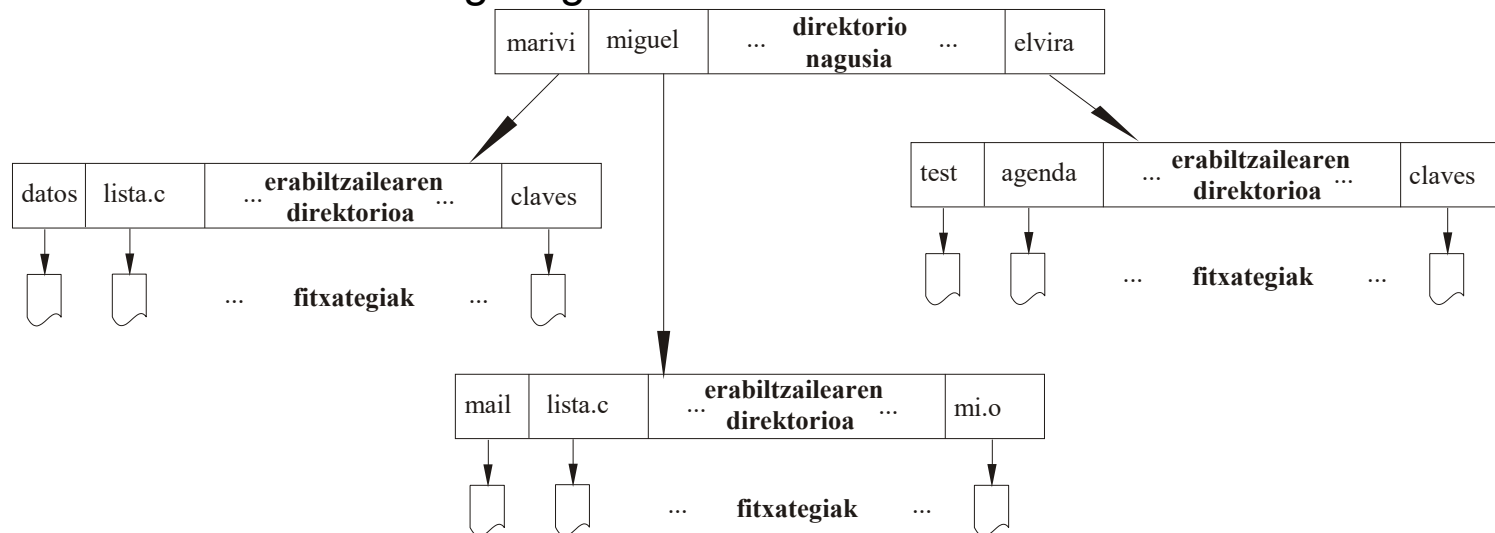


6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



- Bi mailako antolaketa
 - Fitxategi kopuru mugatuaren arazoa konpontzen du
 - Erabiltzaile bakoitzak bere direktorio propioa du
 - Taldekatzeko eta partekatzeko arazoak
 - Konponbidea: Sistema erabiltzaile baten sorrera, hor partekatu daitezkeen fitxategiak gehituz



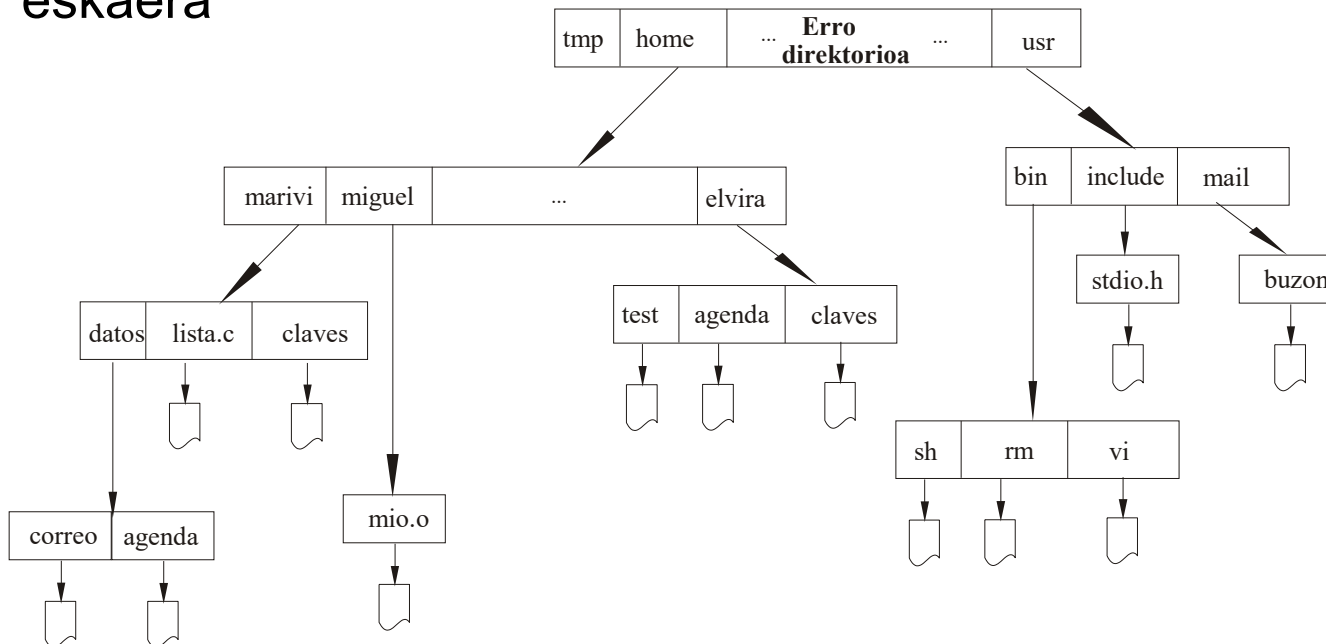
- Biltegiatze gaitasunaren handiagotzeak, fitxategi kopuruaren handiagotzea sortu zuen, berriz izendatzeko arazoak sortuz

6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



- Arbola antolaketa
 - Bilaketa eta taldekatze eraginkorrak
 - Zuhaitzaren egitura guztia erro direktorio batetatik abiatzen da, fitxategi bakoitzarentzat *path* **bakarra** egonik
 - Izen erlatibo eta absolutuak → Laneko direktorioa
 - Direktorio bat hutsik ez badago, ezabatzerako baieztapen eskaera

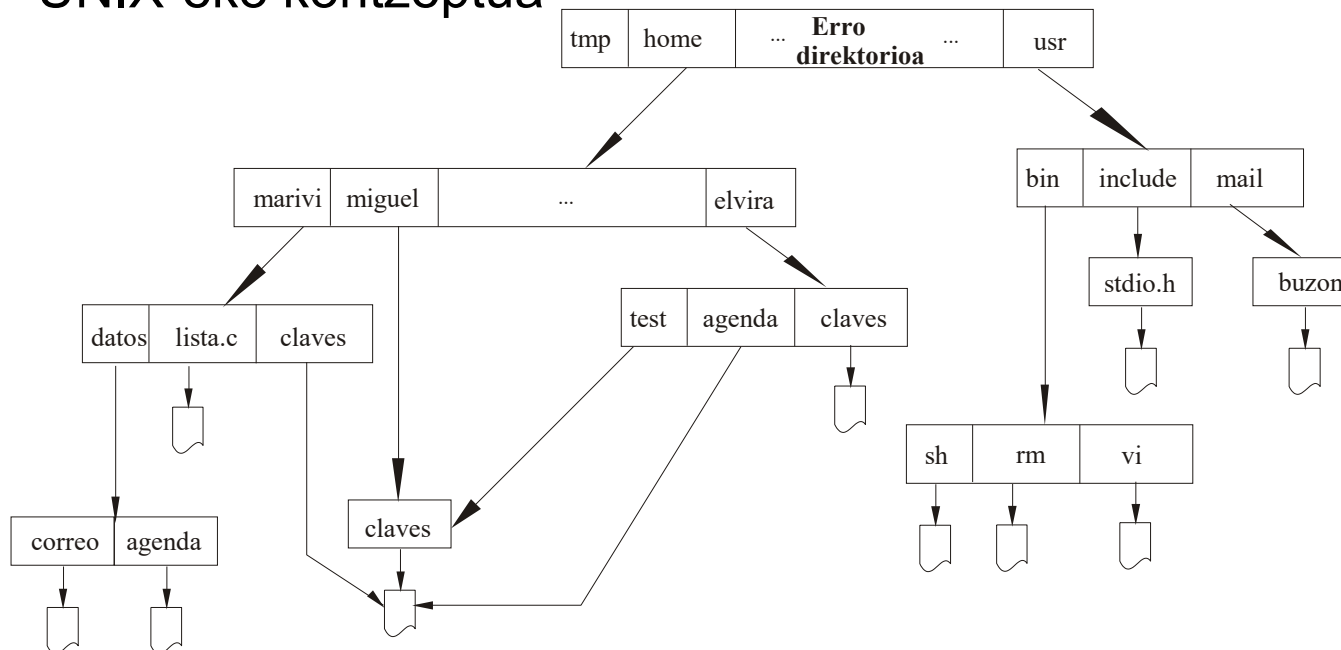


6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



- Grafo aziklikoa
 - Beharrizana: Fitxategiak eta direktorioak partekatu erabiltzaile desberdinen artean
 - Fitxategi batera iristeko *path* bat baino gehiago
 - UNIX-eko kontzeptua



6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



- Grafo aziklikoa
 - **link**: Fitxategi bera izen desberdinekin → esteken kontrola
 - **Esteka fisikoa** (*hard link*): fitxategi biek i-nodo berdinerara dute lotura
 - i-nodoan eremu bat gehitu beharra: esteken kontagailua
 - **Esteka sinbolikoa** (*soft link*): fitxategi berriak, jatorrizko fitxategiari bidea du gordeta
 - Desberdintasunak
 - Jatorrizko fitxategiaren ezabatzea: esteka fisikoaren kasuan, fitxategiari sarbidea du oraindik.
 - Direktorioak: ezin dira direktorioetara zuzenduriko esteka fisikorik sortu
 - Informazio gehigarria:
<https://geekland.eu/que-son-para-que-sirven-enlaces-duros-y-simbolicos/>

6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



- Grafo aziklikoa
 - Estekak ezabatu **unlink**:
 - Esteka fisikoaren kasuan
 - kontagailua dekrementatu
 - baldin kontagailua = 0 → fitxategia ezabatu
 - esteka bakarrik ezabatu eta gainerakoak mantendu
 - Arazoa: Arbolan begiztak agertzea: Bakarrik esteka sinbolikoen kasuan. Irtenbideak:
 - Bakarrik onartu estekak fitxategietara, ez azpidirektorioetara
 - Esteka bat egiten denean, begiztak bilatzeko algoritmo bat
 - UNIX-eko implementazioko muga: bakarrik estekak fitxategi-sistema beraren barruan

6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



- Izen hierarkikoak
 - **Izen absolutua:** izena erro direktoriotik abiatuta adieraziz
 - “/” UNIX, “\” Windows
 - **Izen erlatiboa:** erroa ez den beste direktorio batetik adieraziz
 - pwd
 - Direktorio bereziak:
 - “.” Laneko direktorioa.
 - “..” Aita direktorioa.
 - HOME direktorioa: erabiltzailearen oinarri direktorioa

6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



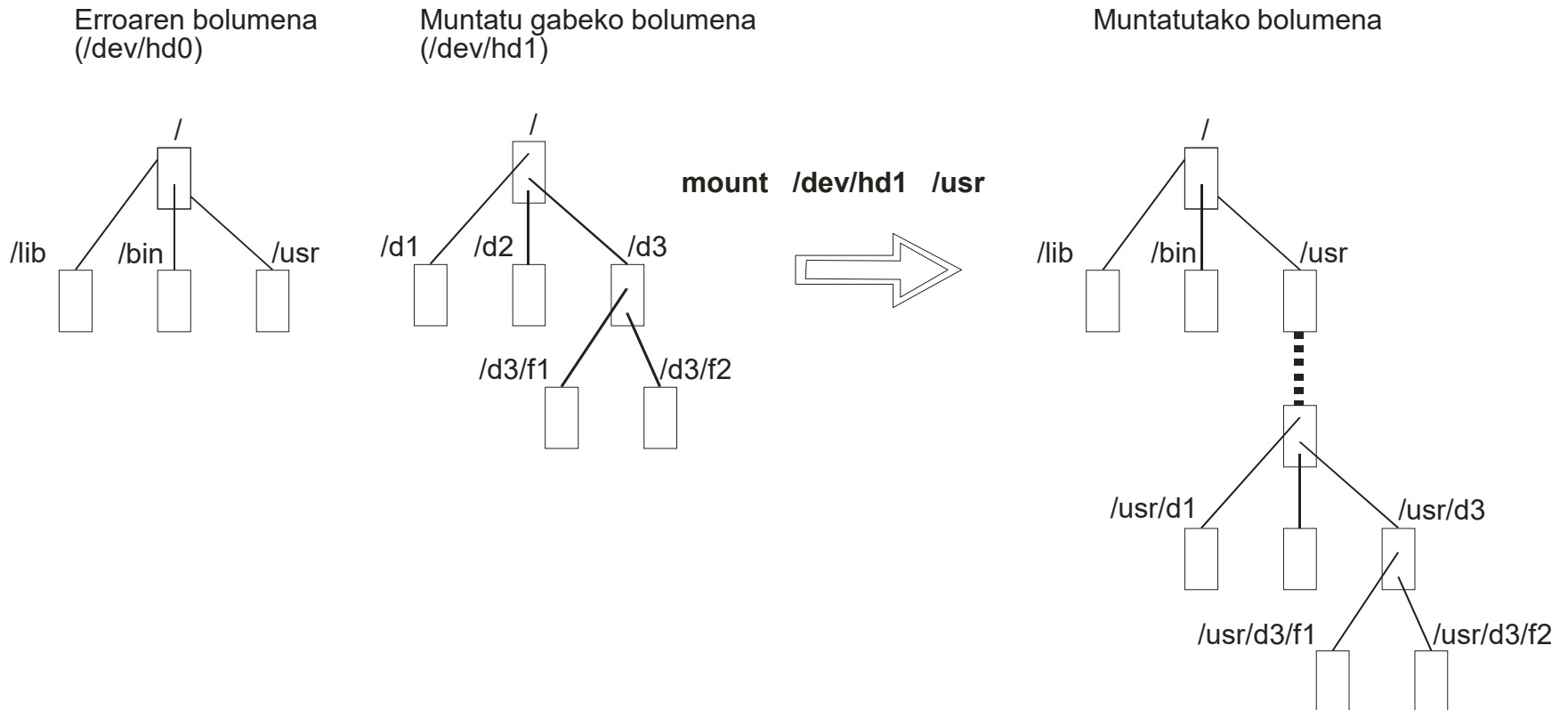
- Direktorioen hierarkiaren antolaketa
 - Direktorio arbola bakarra?
 - Bana gailu logiko bakoitzeko Windowsen (c:\users\miguel\claves, j:\pepe\tmp, ...), bolumenak deitzen zaie
 - Bat sistema osorako UNIXen (/users/miguel/claves, /pepe/tmp, ...)
 - Hierarkia eraikitzeko zerbitzuak behar dira: mount eta unmount.
 - mount /dev/hda /users
 - unmount /users
 - Abantailak: sistemaren irudi bakar bat eta gailua ezkutatzea
 - Desabantailak: izenen itzulpena zaildu egiten da, fitxategien arteko esteka fisikoak egiteko arazoak
 - Ezin dira esteka fisikoak sortu fitxategi-sistema desberdinen fitxategien artean

6. FITXATEGI-SISTEMA

6.3 DIREKTORIOAK



- Direktorioen hierarkiaren antolaketa
 - Bolumenak muntatu eta desmuntatu



6. FITXATEGI-SISTEMA

6.4 POSIX-EN DEFINITURIKO FITXATEGIEN ZERBITZUAK



- Ikuspegi logikoa: byte-sekuentzia
- Erakuslearen kokapena, hortik aurrera eragiketak egiteko
- Fitxategi-deskribatzaileak: zenbaki osoak [0,64K]
- Aurredefinituak (gailuarekiko programa independenteak):
 - fd = 0: sarrera estandarra
 - fd = 1: irteera estandarra
 - fd = 2: erroreen irteera
- fork(): PKB-a bikoizten da, baina prozesuek fitxategi-deskribatzaileen taula partekatzen jarraitzen dute
- exec(): prozesuek fitxategi-deskribatzaileen taula partekatzen jarraitzen dute
- Fitxategiaren atributuak kontsultatu eta aldatzeko zerbitzuak.

- Babesa:

jabea

rwX

taldekoak

rwX

gainerakoak

rwX

6. FITXATEGI-SISTEMA

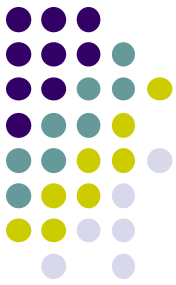
6.4 POSIX-EN DEFINITURIKO FITXATEGIEN ZERBITZUAK



- Fitxategi bat sortu:
`#include <sys/types.h>`
`#include <sys/stat.h>`
`#include <fcntl.h>`
`int creat(char *name, mode_t mode);`
- Fitxategi bat ezabatu:
`int unlink(const char* path);`
 - Esteken kontagailua dekrementatu. Baldin 0, fitxategia ezabatu eta bere baliabideak askatu
- Fitxategi bat ireki:
`int open(char *name, int flag, ...);`
 - Adibideak:
`fd = creat("datos.txt", 0751);`
`fd = open("datos.txt", O_WRONLY|O_CREAT|O_TRUNC, 0751);`

6. FITXATEGI-SISTEMA

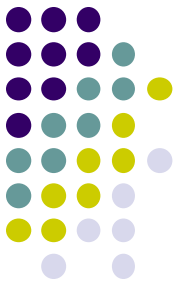
6.4 POSIX-EN DEFINITURIKO FITXATEGIEN ZERBITZUAK



- Fitxategia itxi:
`int close(int fd);`
- Fitxategiko datuak irakurri:
`ssize_t read(int fd, void *buf, size_t n_bytes);`
- Fitxategian datuak idatzi:
`ssize_t write(int fd, void *buf, size_t n_bytes);`
- Fitxategiaren erakuslearen kokapena finkatu:
`off_t lseek(int fd, off_t offset, int whence);`
- Fitxategiaren ezaugarriak aldatu:
`int fcntl(int fildes, int cmd /* arg*/ ...);`

6. FITXATEGI-SISTEMA

6.4 POSIX-EN DEFINITURIKO FITXATEGIEN ZERBITZUAK



- Fitxategiaren ezaugarriak galdetu:

int **stat**(char *name, struct stat *buf);

- fitxategia path bidez identifikatzen da

int **fstat**(int fd, struct stat *buf);

- fitxategia irekita egon behar da

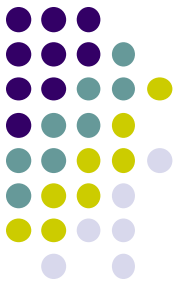
```
struct stat {  
    mode_t st_mode;        /* fitxategiaren modua */  
    ino_t st_ino;          /* fitxategiaren zenbakia */  
    dev_t st_dev;          /* gailua */  
    nlink_t st_nlink;      /* esteka kopurua */  
    uid_t st_uid;          /* jabearen UID */  
    gid_t st_gid;          /* jabearen GID */  
    off_t st_size;         /* byte kopurua */  
    time_t st_atime;       /* azken atzipena */  
    time_t st_mtime;       /* azken eduki-aldaketaren data */  
    time_t st_ctime;       /* ezaugarrien azken aldaketaren data */  
};
```

- Fitxategiaren modua:

S_ISDIR(s.st_mode)	Direktoria bada
S_ISCHR(s.st_mode)	Karaktereko gailua bada
S_ISBLK(s.st_mode)	Blokeko gailua bada
S_ISREG(s.st_mode)	Fitxategi arrunta bada
S_ISFIFO(s.st_mode)	FIFO edo hodi izenduna bada

6. FITXATEGI-SISTEMA

6.4 POSIX-EN DEFINITURIKO FITXATEGIEN ZERBITZUAK



- Fitxategiaren deskribatzailea bikoiztu:
`int dup(int fd);`
- Fitxategiaren luzera moztu:
`int ftruncate(int fd, off_t length);`
- Azken atzipen- eta aldaketa-datak aldatu:
`int utime(char *name, struct utimbuf *times);`
- Besteak:
pipe, mkfifo, link, rename...

6. FITXATEGI-SISTEMA

6.5 DIREKTORIOEN ZERBITZUAK



- Ikuspegi logikoa: izen logiko eta i-nodo bikoteen taula bat
 - Taulako sarrerek honako itxura dute:

```
struct dirent {  
    ino_t d_ino;      /*i-nodoaren zenbakia*/  
    char *d_name;     /* fitxategiaren izena */  
    ...  
}
```
 - Fitxategien izenak luzera aldakorrekoak
- POSIX-ek direktorioak kudeatzeko zerbitzuak eskaintzen ditu

6. FITXATEGI-SISTEMA

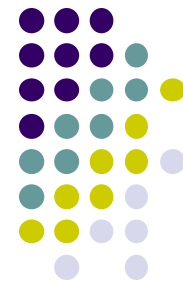
6.5 DIREKTORIOEN ZERBITZUAK



- Direktorio bat sortu:
`#include <sys/types.h>`
`#include <dirent.h>`
`int mkdir(const char *name, mode_t mode);`
- Direktorioa ezabatu:
`int rmdir(const char *name);`
 - Hutsik egon behar da, bestela ezin daiteke ezabatu.
 - Hutsik ez badago, C programatik Shell agindu bat exekutatu daiteke “system()” funtzioa erabiliz
 - `int system(const char *command);`
 - `system("rmdir -rf direktorioa");`
- Direktorioa ireki:
`DIR *opendir(char *dirname);`
 - Direktorioko lehengo sarreran kokatzen da DIR *

6. FITXATEGI-SISTEMA

6.5 DIREKTORIOEN ZERBITZUAK



- Direktorioa itxi:
`int closedir(DIR *dirp);`
- Sarrera bat irakurri:
`struct dirent *readdir(DIR *dirp);`
 - Amaieran badago edo errorea egonez gero NULL
- Lehengo sarrerara joan:
`void rewindir(DIR *dirp);`
- Esteka fisiko edo sinbolikoa sortu:
`int link(const char *existing, const char *new);`
 - Fitxategi-sistema berekoak izan behar dira`int symlink(const char *existing, const char *new)`
 - Ez dute fitxategi-sistema berekoak izan behar
- Direktorioekin ere posible den arren, fitxategiekin bakarrik.

6. FITXATEGI-SISTEMA

6.5 DIREKTORIOEN ZERBITZUAK



- Sarrera bat ezabatu:
`int unlink(char *name);`
 - Esteken kontagailua zerora iritsiz gero, fitxategiak hartutako baliabideak askatu egiten dira eta ezin da berriz atzitu
- Laneko direktorioa aldatu:
`int chdir(char *name);`
- Sarrera (fitxategi) baten izena aldatu:
`int rename(char *old, char *new);`
- Laneko direktorioa zein den jakin:
`char *getcwd(char *buf, size_t size);`
 - buf direktorioaren izena kopiatzeko bufferra eta size bufferraren luzera.
 - buf-en erakuslea itzultzen du

6. FITXATEGI-SISTEMA

6.5 DIREKTORIOEN ZERBITZUAK



- Direktorio bateko sarrerak erakutsi POSIX zerbitzuak erabiliz:

```
#include <sys/types.h>
#include <dirent.h>
#include <stdio.h>
#include <error.h>
#include <unistd.h>
#define BUFF_LUZERA 1024

void main (int argc, char **argv)
{
    DIR *dirp;
    struct dirent *dp;
    char buffer[BUFF_LUZERA];

    /* Argumento kopurua egokia dela egiaztatu */
    if (argc != 2) {
        fprintf (stderr, "Erabilera zuzena: nire_ls direktorioa\n");
        exit(1);
    }
    /* Laneko direktorioa lortu eta bistaratu */
    getcwd (buffer, BUFF_LUZERA);
    printf ("Laneko direktorioa: %s \n", buffer);
}
```

6. FITXATEGI-SISTEMA

6.5 DIREKTORIOEN ZERBITZUAK



```
/* Argumentuan jasotako direktorioa ireki */

dirp = opendir (argv[1]);
if (dirp == NULL){
    fprintf (stderr, "Ezin da %s direktorioa ireki\n", argv[1]);
    exit (1);
}
else{
    printf ("%s direktorioko sarrerak: \n", argv[1]);

    /* Sarrera guztiak banan-banan irakurri eta bistaratu */
    while ((dp = readdir(dirp)) != NULL)
        printf ("%s\n", dp->d_name); /* Izena erakutsi */
    closedir (dirp);
}
exit (0);
}
```

- Ariketa: while begizta aldatu, fitxategi izena erabiliz, fitxategiaren luzera, datak, fitxategi-mota, etab erakusteko

6. FITXATEGI-SISTEMA

6.6 FITXATEGI-SISTEMAK



1. Fitxategien inplementazioa
2. Direktorioen inplementazioa
3. Partekatutako fitxategiak
4. Disko-espazioaren kudeaketa
5. Partizioak eta fitxategi-sistemak
6. Fidagarritasuna

6. FITXATEGI-SISTEMA

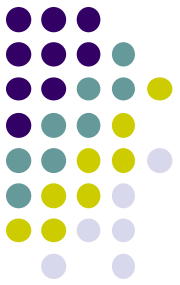
6.6.1 Fitxategien inplementazioa



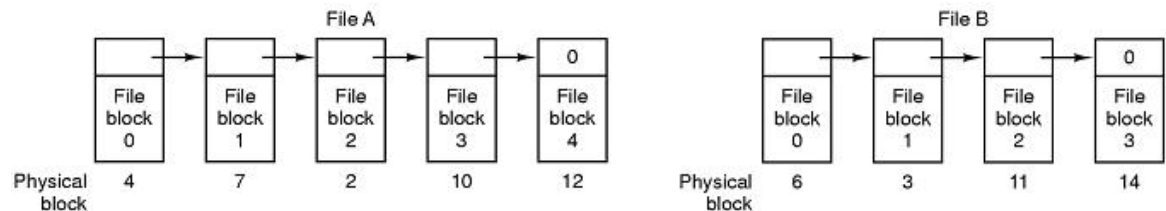
- Fitxategien inplementazioko atazarik garrantzitsuena fitxategi bakoitzari diskoan dagozkion **blokeen jarraipenean** datza
- **Ondoz ondoko blokeen** esleipena:
 - Abantailak
 - Inplementatze erraza. (lehenengo blokea+kopurua)
 - Atzipen eraginkorrak. Bloke asko eragiketa bakarrean
 - Ikuspegi logiko eta fisikoaren arteko paralelismoa
 - Desabantailak
 - Fitxategia sortzen den momentuan fitxategiaren luzera maximoa ezagutzea zaila da → Kanpo-zatiketa.
- Ez da erabiltzen. Fitxategiari esleitzen zaizkion blokeak sakabanatuta egoten dira. Nola lotu orduan ikuspegi logikoa eta fisikoa?

6. FITXATEGI-SISTEMA

6.6.1 Fitxategien inplementazioa



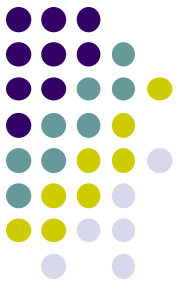
- Blokeen **zerrenda kateatuak**:
 - Fitxategia → diskoko blokeen zerrenda kateatu bat
 - Lehenengo hitzak → Hurrengo blokea erakusten du.
 - Blokeko gainontzeko hitzak → datuak.



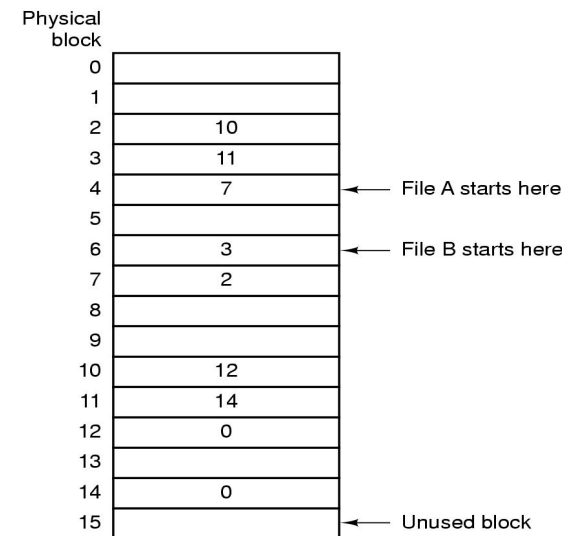
- Abantailak
 - Zatiketa desagertu egiten da
 - Direktorioko sarrerara → lehenengo blokearen helbidea
- Desabantailak
 - Ausazko atzipena astuna da
 - Blokeko datu okupazioa ez da oso eraginkorra (S/I eragiketak 2ren potentzietara normalizatuta daude)
 - Fitxategiaren bloke baten galerak bere osteko bloke guztien galera eragingo du

6. FITXATEGI-SISTEMA

6.6.1 Fitxategien inplementazioa



- **Indizedun zerrenda estekatua (FAT taula):**
 - MS-DOS-en fitxategi – sistema.
 - Aurreko metodoaren desabantaila biak saihas daitezke bloke bakoitzetik erakusleari dagokion hitza atera eta memorian taula edo indize batean gordez gero.
 - Fitxategi bat sortzen denean, bere deskribatzailean fitxategiaren lehen blokeari dagokion deskriptorea gordetzen da.
 - Taularen sarrera bakoitzak bere barnean izan dezake:
 - Disko helbide bat
 - Blokea libre egotearen adierazlea
 - Akatsdun blokearen adierazlea
 - EOF marka



6. FITXATEGI-SISTEMA

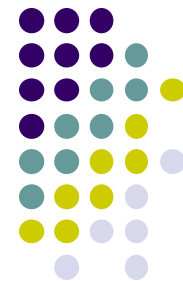
6.6.1 Fitxategien inplementazioa



- **Indizedun zerrenda estekatua (FAT taula):**
 - Abantailak
 - Bloke osoa datuetarako erabilgarri.
 - Ausazko atzipena azkarragoa da, memoria nagusian dagoen taulan bloke guztien helbideak hasieratik ezagutzen baitira.
 - Desabantailak
 - Taula memoria nagusian egon behar da eta bere tamaina handia izan daiteke.
 - Erdibideko konponbidea: Blokeak taldeka batu (Cluster)
 - Handiegia: Barne-fragmentazioa
 - Txikiegia: Taula handiegia eta errendimendu txikiagotzea
 - Konponbidea:
 - Maila anitzeko indizeak
- **Ariketa:** Kalkulatu 1GBko pendrive baten FAT taulak zenbat RAM behar duen blokeak 4KBkoak badira?

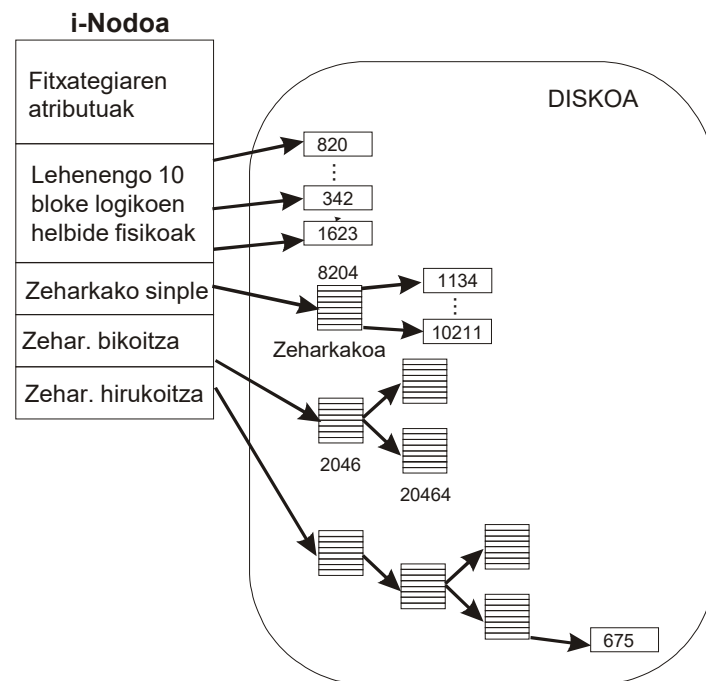
6. FITXATEGI-SISTEMA

6.6.1 Fitxategien inplementazioa



- **i-Nodoak:**

- UNIX-en fitxategi – sistema, ext2-k, ext3-k, ext4-k eta ZFS-k inplementatua.
- Fitxategi bakoitzak i-nodo deituriko datu egitura bana dauka lotuta. i-Nodoak atributuak eta datuen diskoko blokeen helbideak ditu.
- Fitxategi bat zabaltzean dagokion i-nodoa diskotik RAM-era ekartzen da.
- Maila anitzeko indizeak inplementatzen ditu:
 - **Zuzenekoak**, lehenengo 10 bloke logikoei dagozkien bloke fisikoen erakusleak i-Nodoan daude.
 - Nahikoak ez badira, fitxategi handien kasuan, **zeharkako bloke sinpleak** diskoko helbideak biltzen ditu.
 - Nahikoa ez balitz, **zeharkako bloke bikoitzak** zeharkako bloke sinpleak erakusten ditu, eta **zeharkako hirukoitzak** zeharkako bikoitzak.



6. FITXATEGI-SISTEMA

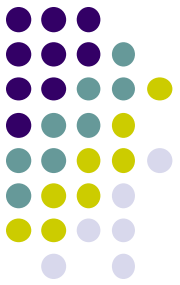
6.6.1 Fitxategien inplementazioa



- Ariketa: Disko bateko blokeak 1KBkoak dira eta guztira 16384 bloke ditu. Fitxategi baten 53600 kokapenean dagoen bytea atzitu nahi dugu. Atzitu beharreko fitxategiaren eta fitxategi sistemaren kontroleko informazio guztia memorian dagoela suposatuz, zenbat atzipen egin beharko dira diskora, kasurik txarrean:
 1. FAT motako fitxategi sistema bat erabiltzen bada?
 2. UNIX motako fitxategi sistema bat erabiltzen bada?
 3. Espazioa kudeatzeko esteka sinplez lotutako bloke kateatuak erabiltzen duen balizko fitxategi sistema baten kasuan?

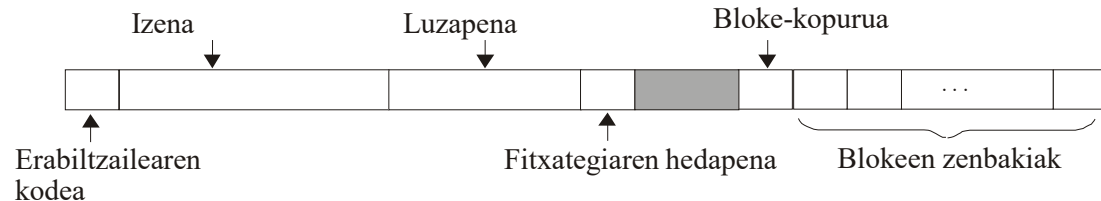
6. FITXATEGI-SISTEMA

6.6.2 Direktorioen inplementazioa



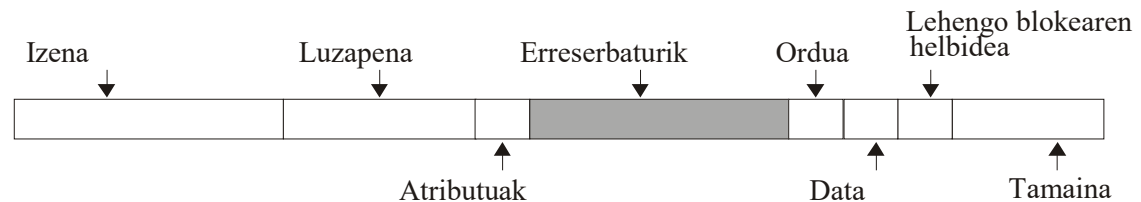
- Direktorio sistemaren funtzio nagusia fitxategiaren ASCII bide-izenetik fitxategiaren datuak lortzeko beharrezko informaziora itzultzea da
- CP/M (ZAHARKITUA): direktorio bakarra

- Sarrerak



- MS-DOS: hierarkikoa

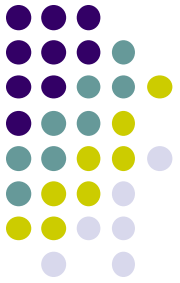
- Sarrerak



- 1. blokearen zenbakia indize bezala erabiltzen da FAT edo indizedun zerrenda kateatuko taulan. Hortik 2. blokeera pasa daiteke, 3.era...

6. FITXATEGI-SISTEMA

6.6.2 Direktorioen inplementazioa



- UNIX: hierarkikoa
 - Sarrera:



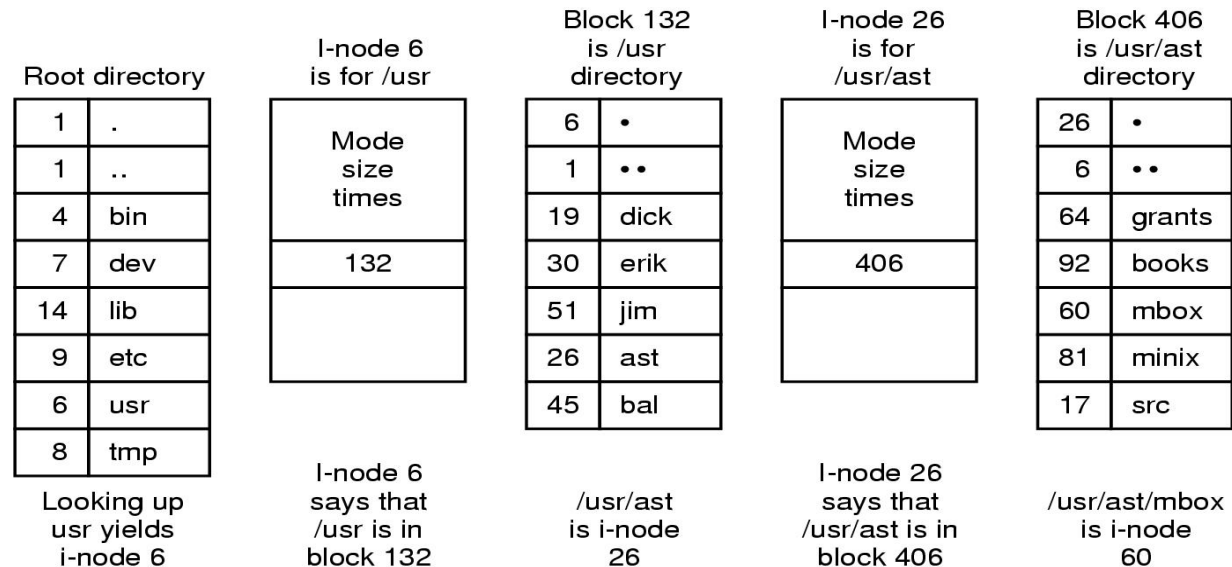
- Fitxategien atributu guztiak (mota, tamaina, data...) i-nodoetan daude
- i-nodo datu-egiturak diskoan gordeta daude, fitxategi-sistemarako dagoen zonaldean eta zenbakiz ordenatuta



6. FITXATEGI-SISTEMA

6.6.2 Direktorioen inplementazioa

- Bide-izenaren itzulpena
 - Adibidea `/usr/ast/mbox` bilatzeko pausuak:
 - Fitxategia aurkitu da eta fitxategiari dagokion i-nodoa itzultzen du
 - Ez du fitxategia aurkitu eta path-eko azken azpi-direktorioan gaude
 - Path-eko direktorio batean gaude eta ez daukagu baimenik azpi-direktoriara sartzeko



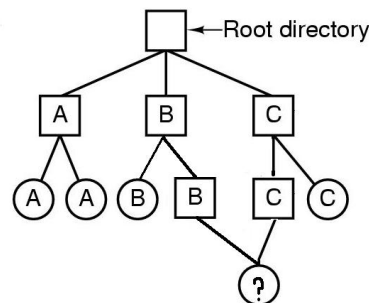
- Direktorio guztietan bi sarrera daude gutxienez
- `'.'` eta `'..'`, laneko eta aita edo sorburu i-nodoak dira.

6. FITXATEGI-SISTEMA

6.6.3 Partekatutako fitxategiak



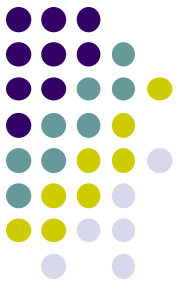
- Partekatutako fitxategi eta direktorioen arteko loturari “link” edo esteka deritzo
 - Fitxategi berari izen bat baino gehiago ematea ahalbidetzen du



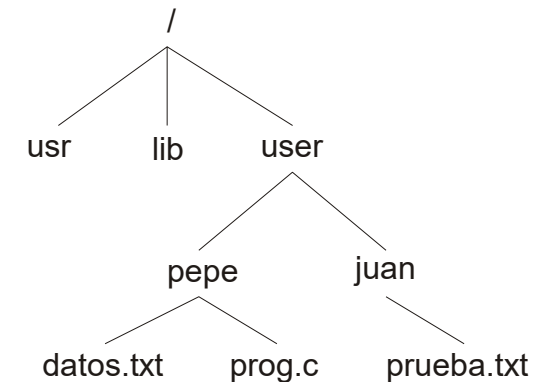
- Zenbait arazo sortzen ditu:
 - Direktorioek (CP/M) diskoko helbideen kopia bat badute, eta artxiboren bat aldatu egiten bada bloke berriak sartuz, aldaketa hau ez da ikusten beste artxiboan.
 - Irtenbideak:
 - i-nodoak erabiltzea (UNIX-en). *Hard link* edo esteka fisikoa.
 - *Soft link* edo esteka sinbolikoak erabiltzea.

6. FITXATEGI-SISTEMA

6.6.3 Partekatutako fitxategiak



- Esteka fisikoa

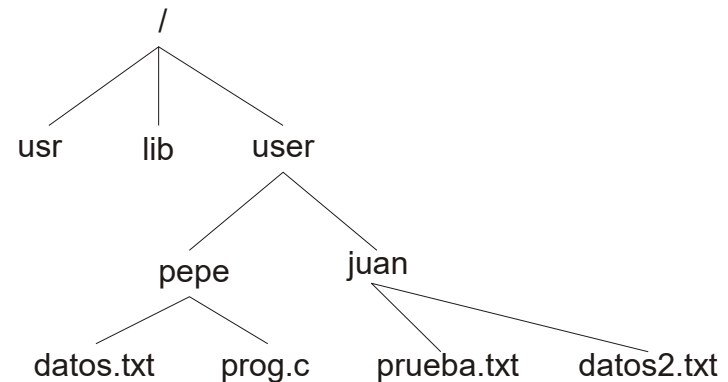


pepe

▪	23
..	100
datos.txt	28
prog.c	400

juan

▪	80
..	100
prueba.txt	60



pepe

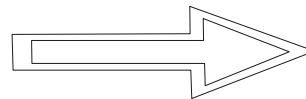
▪	23
..	100
datos.txt	28
prog.c	400

juan

▪	80
..	100
prueba.txt	60
datos2.txt	28

28 i-nodoa

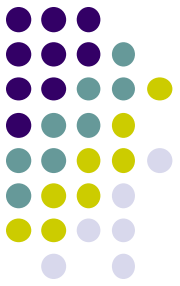
estekak = 2
fitxategiaren
edukia



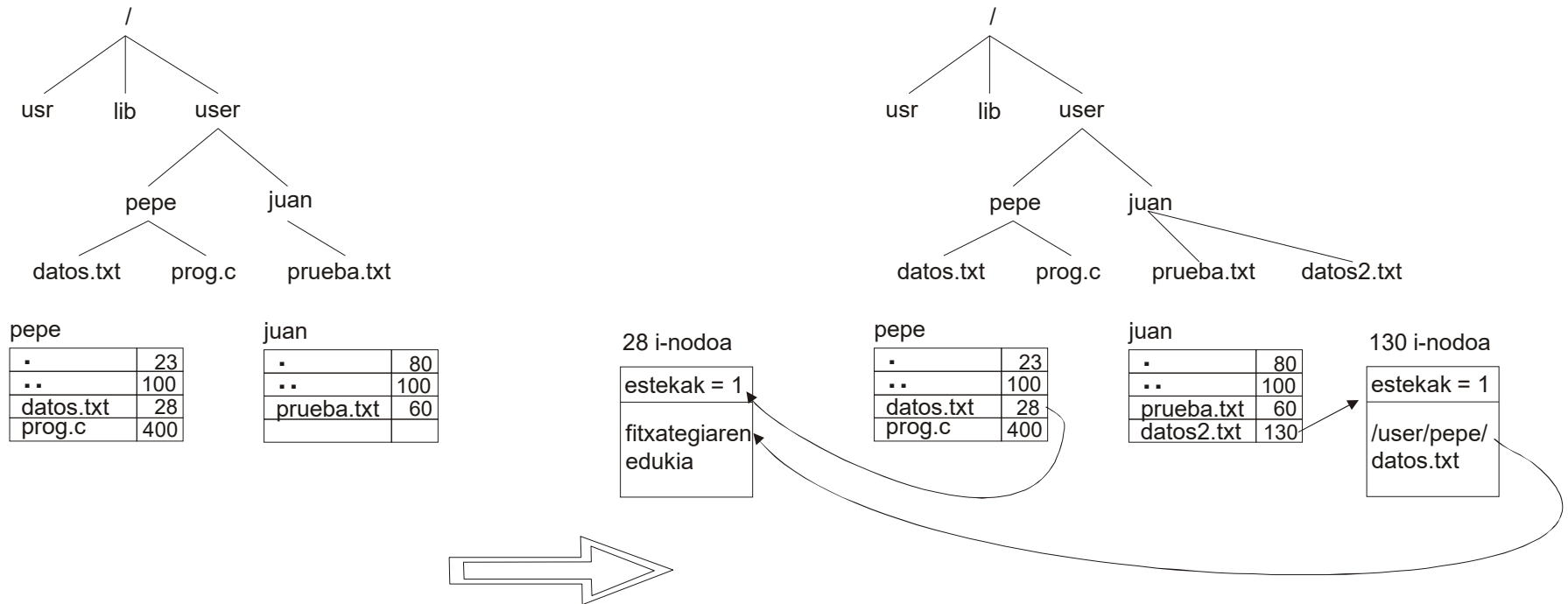
In /user/pepe/datos.txt /user/juan/datos2.txt

6. FITXATEGI-SISTEMA

6.6.3 Partekatutako fitxategiak



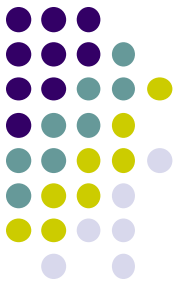
- Esteka sinbolikoa



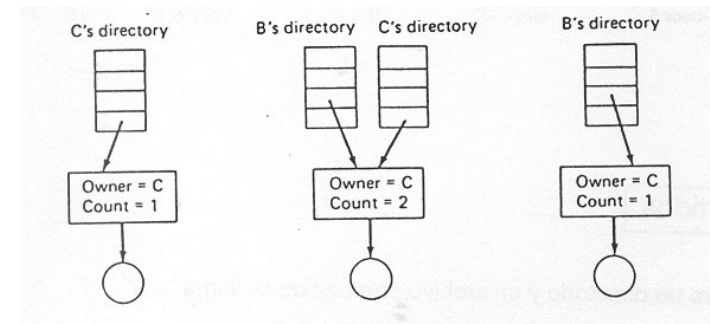
In -s /user/pepe/datos.txt /user/juan/datos2.txt

6. FITXATEGI-SISTEMA

6.6.3 Partekatutako fitxategiak

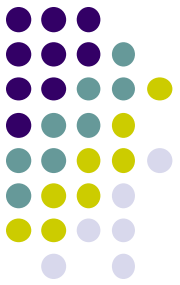


- Esteken mugak
 - Esteka fisikoa
 - Fitxategia ez da ezabatzen esteka kopurua 0 izan arte
 - Bolumen bereko fitxategietara, ez direktorioetara
 - Ez da aldatzen fitxategiaren jabea
 - Esteka sinbolikoa
 - Esteka fisikoa ezabatuz gero...
...esteka sinbolikoak ez du balio
 - Fitxategiekin eta direktorioekin egin daiteke
 - Bolumen eta makina desberdinen artean egin daiteke

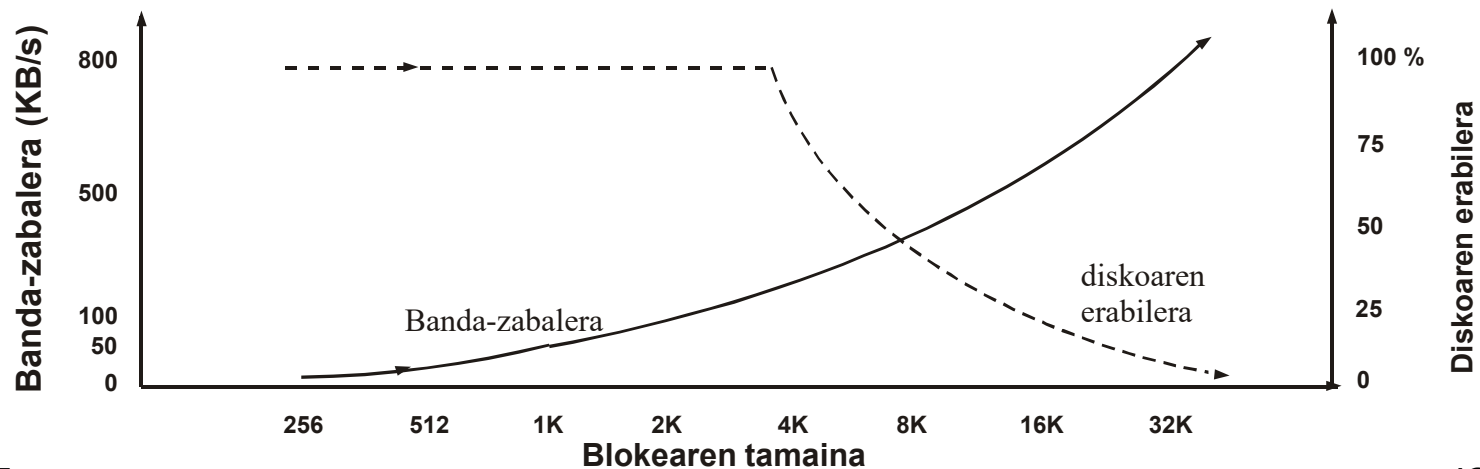


6. FITXATEGI-SISTEMA

6.6.4 Disko-espazioaren kudeaketa

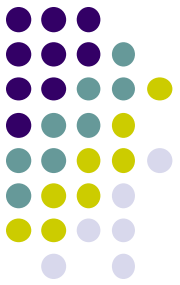


- Esleipen estrategiak
 - Ondoz ondoko blokeak. Kanpo-zatiketa dela eta baztertua.
 - Bloke banatuetan. Ondoz ondokoak izan daitezke edo ez.
- Blokeen tamaina
 - Handia bada → Diskoaren erabilera ez eraginkorra, barne-zatiketa
 - Txikia bada → Bloke asko → Atzipen astuna, S/I eragiketa gehiegi
 - Oreka: Atzipen denbora ↔ Espazioaren erabilpena 4KB-8KB
 - Goranzko joera: Multimedia gehiago, disko merkeagoak, handiagoak

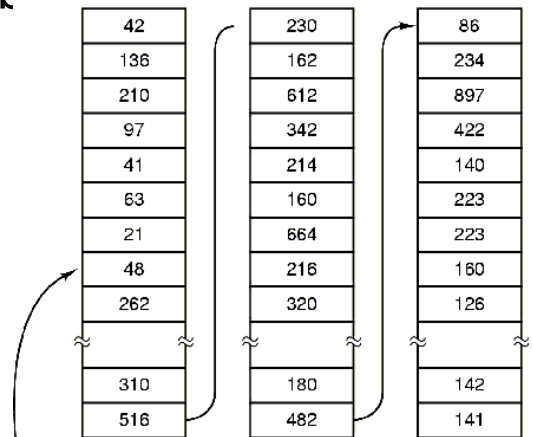


6. FITXATEGI-SISTEMA

6.6.4 Disko-espazioaren kudeaketa



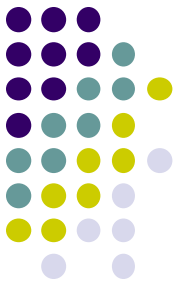
- Bloke libreen jarraipena
 - FAT
 - Zerrenda estekatuen bidez
 - Bit-mapen bidez
- Zerrenda estekatuak
 - Diskoko zenbait blokek libre dauden diskoko blokeen zenbakiak gordetzen dituzte. Adib:
 - 1KBeko blokeak eta 32biteko indizeak
 - 20MB disko(txo) batek 80 bloke behar ditu
 - Diskoa bete heinean gutxiago behar da
 - Aldaera: Zerrendan zein bloke dagoen libre + zenbat bloke jarraian dauden libre adierazi



A 1K disk block can hold 256
32-bit disk block numbers

6. FITXATEGI-SISTEMA

6.6.4 Disko-espazioaren kudeaketa



- Bit-mapak

- n blokeko disko batek → n biteko mapa behar du.
- Bit mapak beti berdin okupatzen du (aurrekoak baino gutxiago)
- Espazioa libre badago 1 balioa izango du, okupatuta badago 0.

1001101101101100
0110110111110111
1010110110110110
0110110110111011
1110111011101111
1101101010001111
0000111011010111
1011101101101111
1100100011101111
~ ~
0111011101110111
1101111101110111

A bit map

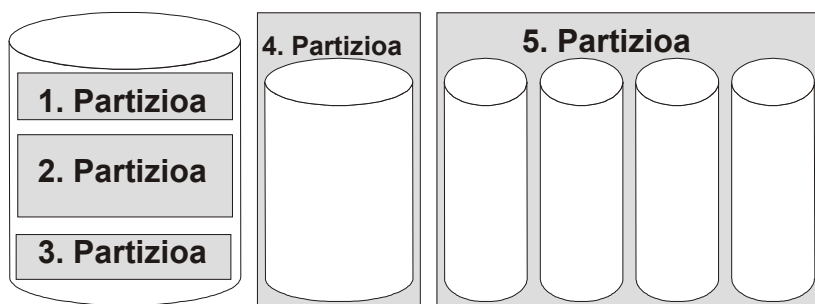
- Blokeak kudeatzeko ez ezik, esleitutako i-nodoak eta libre daudenak kudeatzeko ere erabil daitezke bit-mapak

6. FITXATEGI-SISTEMA

6.6.5 Partizioak eta fitxategi-sistemak



- Fitxategi-sistema batek, biltegitratze sekundarioko gailu baten informazioa antolatzen du, SE-ei ulergarri zaien formatuan
- Diskoetan fitxategi-sistema instalatu aurretik beharrezkoa izaten da zatitzea **partizio** edo **bolumen**etan.
- **Partizioa** identitate propioa duen diskoaren atal bat da. SE-tik unitate logiko independente moduan kudea daiteke.



- Hiru partizio dituen diskoa
- Partizio bakarra duen diskoa
- Zenbait diskoetara hedatutako partizioa

6. FITXATEGI-SISTEMA

6.6.5 Partizioak eta fitxategi-sistemak



- Fitxategi-sistema/gailu erlazioa:
 - Ohikoa: 1 gailu N fitxategi-sistema (partizio)
 - Fitxategi erraldoiak: N gailu 1 fitxategi-sistema
- Partizio-taulak partizio bakoitzaren hasiera, tamaina eta mota adierazten ditu.
- Partizioak sortutakoan, SE-ak **fitxategi-sistemaren** egiturak antolatu behar ditu partizioen barruan. `format` edo `mkfs` moduko aginduak eskaintzen dizkio erabiltzaileari. Adib:
 - `#mkswap -c /dev/hda2 20800`
 - `#mkfs -c /dev/hda3 -b 8192 123100`
 - -c hondatutako blokeak aztertzen ditu
 - 20800 123100 partizioen tamainak blokeetan
 - -b 8192 fitxategi-sistema horretako blokeen neurria 8KBetan finkatzeko

6. FITXATEGI-SISTEMA

6.6.5 Partizioak eta fitxategi-sistemak



- **Fitxategi-sistema:** metainformazio autoedukia eta datu-multzo koherenteak. SE-aren kargarako informazioa dauka, fitxategien antolaketarena, eta blokeen okupazioaren gainekoa.

MS-DOS



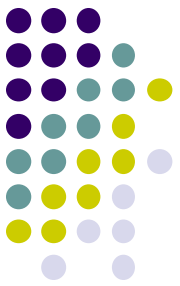
UNIX



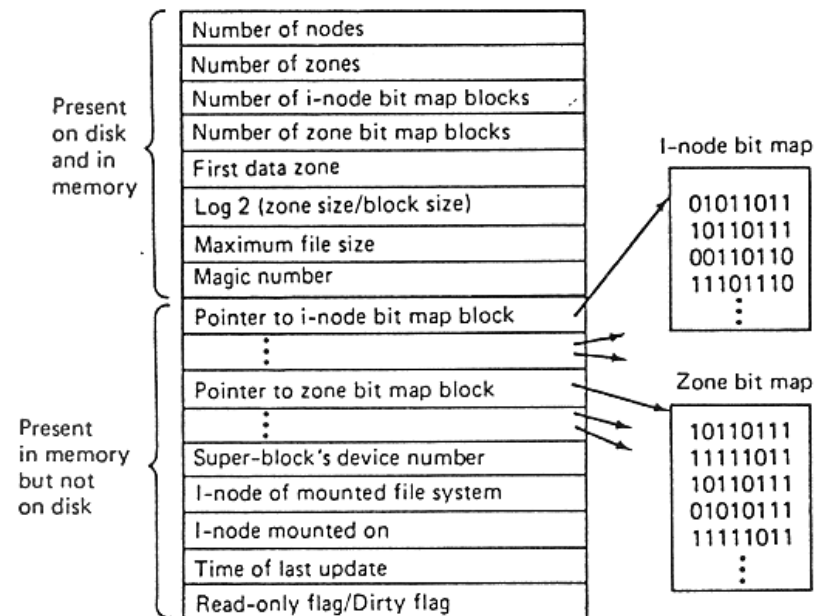
- **Boot** edo **abio-blokea**, fitxategi-sistema guztietan. Fitxategi-sistemak SE bat dakarren ala ez adierazten du
 - ROM-eko abiarazte programak exekutatu du
 - SE-a memorian kargatzen du
- **Superblokeak**, FS-ren antolaketari buruz kalkula ez daitekeen informazioa dakar
- **Bi bit-mapa:** bata blokeen jarraipenerako, bestea i-nodoena
- **i-Nodoak:** Fitxategien atributuekin, eta esleitutako bloke fisikoekin
- **Datuak** (fitxategiak, direktorioak, zeharkako blokeak) gordetzeko blokeak. Lehengoan, erro direktorioa

6. FITXATEGI-SISTEMA

6.6.5 Partizioak eta fitxategi-sistemak



- **Superblokea:** FSren egiturari buruzko informazioa du (FSren 6 arloen tamainaren informazioa ematea da betebehar nagusia)
- Superblokea hondatuz gero FS osoa galtzen da
- Diskoaren tamaina, blokeen tamaina eta i-nodo kopurua emanda, erreza da bloke eta i-nodoen bit-maparen tamaina eta i-nodoen bloke kopurua lortzea
- SE arrankatzean, erro dispositiboaren superblokea memoriako taula batera iraultzen da.
- Era berean, beste FS batzuk muntatzean, euren superblokeak ere memoriako superblokeen taulan kargatzen dira.
- Adibidea: MINIXeko superblokea



6. FITXATEGI-SISTEMA

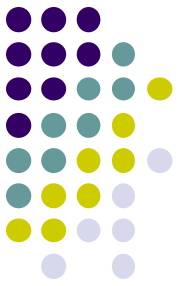
6.6.5 Partizioak eta fitxategi-sistemak



- **Ariketa:** 20 GB-ko disko batean UNIX motako sistema eragileko fitxategi sistema bat daukagu. Blokeak 1KB-ekoak dira. Superblokeko informazioak 50.000 i-nodo daudela dio. i-nodoek 64 byte betetzen dituzte eta fitxategiaren atributuez aparte, kasu honetan 7 bloke erakusle zuzen dauzkate, zeharkako bloke erakusle sinple bat eta beste zeharkako erakusle bikoitz bat ere bai. Bloke baten indizeak 4 bytekoak dira. Bai i-nodoak eta bai datu blokeak goranzko ordenenean ematen dira (0 i-nodoa okupatuta bezala markatuta dago beti). Direktorioetako sarrerek 16 byte okupatzen dituzte: 2 i-nodoa adierazteko eta 14 izenerako eta direktorio bat sortzen denean, bi sarrera izaten ditu beti “.” eta “..”. Zera eskatzen da:
 - Lehenengo datu blokea zein izango den zehazteko egin behar diren kalkuluak atera.
 - Diskoa formateatu ostean 2.097.152 byteko /Kixote.txt fitxategia sortzen da (2.048 datu bloke). Adierazi zeintzuk izango diren fitxategiaren ondoko byteak gordeko dituzten blokeak: 1, 5.170, 41.000 eta 746.500 byteak.

6. FITXATEGI-SISTEMA

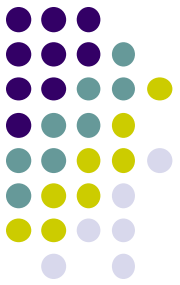
6.6.5 Partizioak eta fitxategi-sistemak



- **Enuntziatutik lortutako datuak:**
 - Disko tamaina: 20 GB
 - Fitxategi-sistema: UNIX motakoa
 - Bloke tamaina: 1KB
 - i-nodo kopurua: 50.000
 - i-nodo bakoitzaren tamaina: 64 byte
 - Bloke erakusle zuzen kopurua: 7
 - Zeharkako bloke erakusle sinple kopurua: 1
 - Zeharkako bloke erakusle bikoitz kopurua: 1
 - Bloke baten indize tamaina: 4 byte
 - Direktorio sarrera tamaina: 16 byte
 - i-nodo zenbakia: 2 byte
 - Fitxategi izena: 14 byte

6. FITXATEGI-SISTEMA

6.6.5 Partizioak eta fitxategi-sistemak



- **Datuetatik lortutako informazioa:**

- Disko tamaina: 20 GB + Bloke tamaina: 1KB
 - Bloke kopurua: 20M
- i-nodo kopurua: 50.000
- i-nodo bakoitzaren tamaina: 64 byte
 - Bloke erakusle zuzen kopurua: 7
 - Zeharkako bloke erakusle single kopurua: 1
 - Zeharkako bloke erakusle bikoitz kopurua: 1
 - Bloke baten indize tamaina: 4 byte
 - Direktorio sarrera tamaina: 16 byte
 - i-nodo zenbakia: 2 byte
 - 2byte = 16 bit
 - $2^{16} = 64k \rightarrow$ beharrezkoa 50.000 i-nodoen indize bezala
 - Fitxategi izena: 14 byte

- i-nodoentzako tamaina:
$$\frac{\text{i-nodo kopurua} \times \text{i-nodo bakoitzaren tamaina}}{\text{Bloke tamaina}} = \frac{50.000 \times 64}{1K} = 3200 \text{ bloke}$$

- i-nodoen bit-mapa tamaina:
$$\frac{\text{i-nodo kopurua} \times (1\text{byte}/8\text{bits})}{\text{Bloke tamaina}} = \frac{50.000/8}{1K} = 6,25 \rightarrow 7 \text{ bloke}$$

- Blokeentzat bit-mapa tamaina:
$$\frac{\text{bloke kopurua} \times (1\text{byte}/8\text{bits})}{\text{Bloke tamaina}} = \frac{20M/8}{1K} = 2500 \text{ bloke}$$

6. FITXATEGI-SISTEMA

6.6.5 Partizioak eta fitxategi-sistemak



- **Ebazpena:**

- Lehenengo datu blokea zein izango den zehazteko egin behar diren kalkuluak atera.

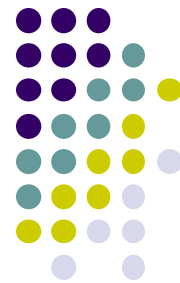
UNIX

Boot	Super-Blokea	Bit-mapak	i-Nodoak			Datuak eta Direktorioak			
------	--------------	-----------	----------	--	--	-------------------------	--	--	--

- Boot bloke tamaina + Super-Blokea tamaina + i-nodo bit-mapa + blokeen bit-mapa + i-nodoak tamaina = $1 + 1 + 7 + 2500 + 3200 = 5709$
- 5709 da lehenengo datu blokea (Blokeak 0tik kontatuz izendatzen direla kontuan hartuz)
- Diskoa formateatu ostean 2.097.152 byteko /Kixote.txt fitxategia sortzen da (2.048 datu bloke). Adierazi zeintzuk izango diren fitxategiaren ondoko byteak gordeko dituzten blokeak: 1, 5.170, 41.000 eta 746.500 byteak.
 - Zein i-nodo zenbakitan gordeko da Kixote.txt?
 - 0 zenbakidun i-nodoa: erro direktoriora esleituta dago beti. Enuntziatuan honela adierazi da hori: 0 i-nodoa okupatuta bezala markatuta dago beti.
 - 1 zenbakidun i-nodoa esleituko zaio Kixote.txt fitxategiari

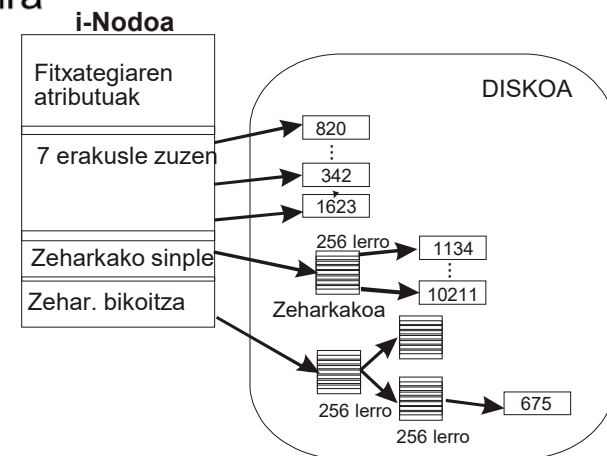
6. FITXATEGI-SISTEMA

6.6.5 Partizioak eta fitxategi-sistemak



● Ebazpena:

- Diskoa formateatu ostean 2.097.152 byteko /Kixote.txt fitxategia sortzen da (2.048 datu bloke). Adierazi zeintzuk izango diren fitxategiaren ondoko byteak gordeko dituzten blokeak: 1, 5.170, 41.000 eta 746.500 byteak.
- OHARRA: Denak 0. posiziotik abiatzen direla kontuan hartuz ebatzita
- Nola erreferentzia ditzaket 2.048 blokeak? (Enuntziatutik abiatuz)
 - Bloke erakusle zuzen kopurua: 7
 - 7 bloke: 0. – 6. Blokeak
 - Zenbat bloke erreferentzia ditzaket zeharkako bloke erakusleekin?
 - Enuntziatua: Bloke baten indizeak 4 bytekoak dira
 - $$\frac{\text{Bloke tamaina}}{\text{indize tamaina}} = \frac{1K}{4} = 256 \text{ bloke erakusle ditu}$$
 - Zeharkako bloke erakusle sinple kopurua: 1
 - 256 bloke: 7. – 262. Blokeak
 - Zeharkako bloke erakusle bikoitz kopurua: 1
 - $256 \times 256 = 65.536$ bloke erakusle izan ditzaket
 - $2.048 - 256 - 7 = 1.785$ bloke erakusle falta dira fitxategi guztia erreferentziatzeko:
 - 263. – 2047. Blokeak
 - Zeharkako bloke erakusle bikoitzetik eratorritako 256 zeharkako bloke erakusle sinpleetatik 7 bakarrik behar dira ($1.785/256 = 7$)



6. FITXATEGI-SISTEMA

6.6.5 Partizioak eta fitxategi-sistemak

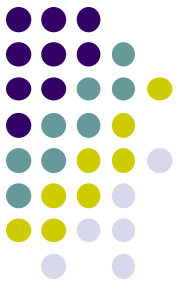


- **Ebazpena:**

- Diskoa formateatu ostean 2.097.152 byteko /Kixote.txt fitxategia sortzen da (2.048 datu bloke). Adierazi zeintzuk izango diren fitxategiaren ondoko byteak gordeko dituzten blokeak: 1, 5.170, 41.000 eta 746.500 byteak.
 - OHARRA: Denak 0. posiziotik abiatzen direla kontuan hartuz ebatzita
 - Zenbat garren bloke logikoan daude byte horiek?
 - 1 -> 0. Blokean
 - 5.170 -> 5. Blokean
 - 41.000 -> 41. Blokean
 - 746.500 -> 746. Blokean
 - Non aurki dezaket bloke horri dagokion bloke fisikoa?
 - 1 -> bloke erakusle zuzenen 0. sarreran
 - 5.170 -> bloke erakusle zuzenean 5. sarreran
 - 41.000 -> Zeharkako bloke erakusle sinpleetatik 34. sarreran
($41 - 7 = 34$)
 - 746.500 -> Zeharkako bloke erakusle bikoitzetik 1. sarrerara loturiko zeharkako bloke erakusle sinpleetako 227. sarreran
($746 - 7 - 256 = 483 - 256 = 227$)

6. FITXATEGI-SISTEMA

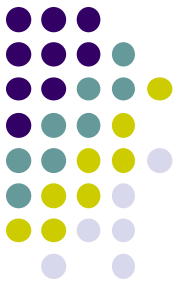
6.7 Fitxategi-zerbitzaria



- Fitxategiak kudeatzeko datu-egiturak
 - **i-nodoen taula**
 - Fitxategiak zabaltzean i-nodo taulara (memoriara) ekartzen da beraien i-nodoa
 - i-nodo taulak diskoan azaltzen diren baino eremu gehiago ditu: i-nodo zenbakia, gailua eta kontagailua (zenbat aldiz dagoen zabalik, prozesuek open egitean ++, close egitean --)
 - kontagailua 0-ra heltzean, taulatik atera eta diskoan gorde egiten da.
 - Taula honen luzerak SE-ak batera kudea dezaken fitxategi-kopuru maximoa finkatzen du

6. FITXATEGI-SISTEMA

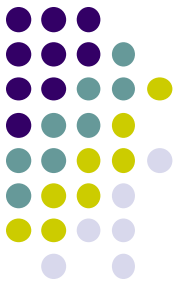
6.7 Fitxategi-zerbitzaria



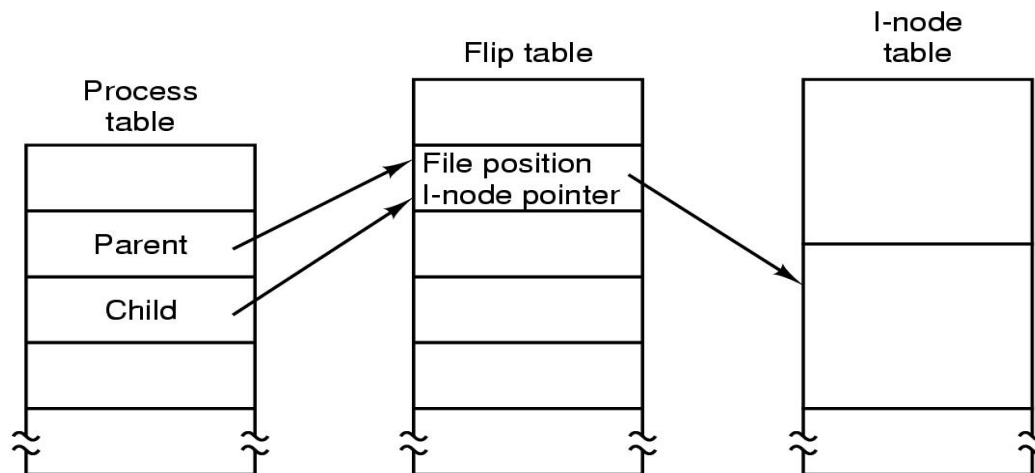
- Fitxategiak kudeatzeko datu-egiturak
 - Prozesuek **irekitako fitxategien taula**
 - Prozesuaren PKB-an
 - Taula honen luzerak prozesu batek batera irekita izan dezakeen fitxategi-kopurua mugatzen du
 - Fitxategi deskribatzailea fd taula honen indizea da
 - Taula goranzko ordenan betetzen da, beti libre dagoen zenbakirik txikiena esleituz
 - UNIX motako sistemetan prozesu bakoitzak hiru deskribatzaile ditu irekita hasieratik:
 - Sarrera estandarra, fd = 0
 - Irteera estandarra, fd = 1
 - Errore estandarra, fd = 2
 - Kokapena finkatzeko hirugarren egitura bat behar da: FILP taula
 - Fitxategi bera partekatuta egon daitekeelako

6. FITXATEGI-SISTEMA

6.7 Fitxategi-zerbitzaria

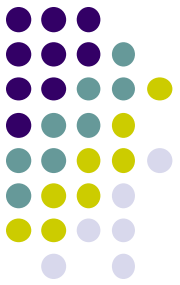


- Fitxategiak kudeatzeko datu-egiturak
 - File In-Line Position **FILP** taula
 - Non gorde partekatutako fitxategien kokapena?
 - i-nodoan? Ez, prozesu guztiek kokapen bera ikusiko lukete.
 - Prozesuen PKB-an? Prozesu independenteentzako balioko luke, baina hierarkia bereko prozesuek partekatutako fitxategiaren kokapen bera ere partekatzen dute
 - Non orduan? PKB eta i-nodoen arteko taula batean



6. FITXATEGI-SISTEMA

6.7 Fitxategi-zerbitzaria



- Fitxategiak/direktorioak kudeatzeko datu-egiturak
 - **Superblokeen taula**
 - Fitxategi-sistema bakoitzak sarrera bana dauka superblokeen taulan. Taulako sarrera bakoitzean:
 - Muntatutako bolumenaren erroaren i-nodoa
 - Muntaia egin den direktorioaren i-nodoa
 - Muntaia egin den direktorioaren i-nodoan, muntatutako superblokearen sarrera jartzen da
 - Sistema berriaren erroaren i-nodoa i-nodoen taulan kargatzen da

