

File yaklaşımı

- Dosya sistemi çoğu kullanıcı için işletim sisteminin görünen kısmıdır.
- Dosya sistemi iki kısımdan oluşur:
 - Dosya kümesi
 - Dizin yapısı
- Bilgisayarlar verileri farklı depolama birimlerinde saklarlar (manyetik diskler, tape, optik diskler, ...).
- İşletim sistemi depolama biriminin fiziksel özelliğini soyutlamak için **file** adı verilen mantıksal depolama birimi tanımlar.
- Dosyalar işletim sistemi tarafından fiziksel cihazlarla eşleştirilirler.
- Depolama birimleri kalıcı veri saklama birimleridir.
- Bir dosya kendi aralarında ilişkili veriler kümesidir.
- Dosyalar, programları veya verileri saklarlar.

3

File yaklaşımı

- Veri dosyaları **nümerik**, **alfabetik**, **alfanümerik** veya **binary** olabilir.
- Dosyalar, **metin dosyaları** veya **formatı belirlenmiş** olabilir.
- Dosyadaki bilginin ne şekilde tanımlanacağı **oluşturan tarafından belirlenir**.
- **Text file**, satırlar halinde organize edilmiş karakter sıralarından oluşur.
- **Source file**, bir grup fonksiyon içerir.
- **Executable file**, bir grup kod bölümünden oluşur ve hafızaya alınarak çalıştırılabilir.
- Bir dosyanın içeriği, **bit**, **byte**, **satır** veya **kayıtlardan** oluşacak şekilde yapılandırılabilir.

4

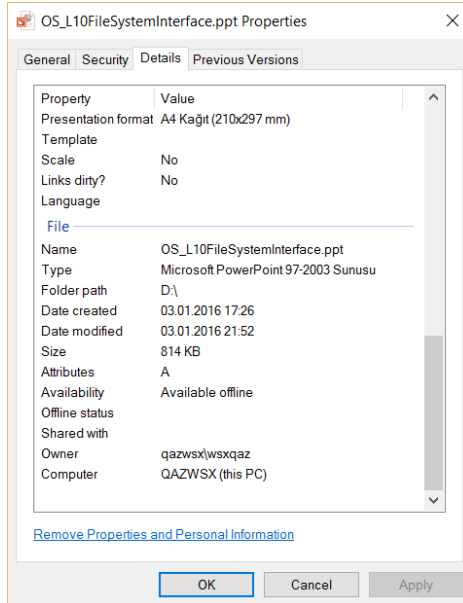
File yaklaşımı

Dosya özellikleri

- Kullanıcıların kolaylıkla işlem yapabilmesi için **her dosyaya isim verilir.**
- Dosyaların özellikleri **işletim sistemlerine göre farklılık gösterir:**
 - **Name:** Kullanıcıların kolaylık okuyabileceği isimdir.
 - **Identifier:** Tekil etikettir, genellikle sayı kullanılır.
 - **Type:** Sistemler için dosyanın türünü belirler.
 - **Location:** Bu bilgi bir cihazı veya cihaz üzerindeki dosyanın yerini işaret eder.
 - **Size:** Dosyanın mevcut boyutunu gösterir.
 - **Protection:** Erişim bilgisidir, dosyaya kimin erişeceği, okuyacağı, çalıştıracağı belirlenir.
 - **Time, date, user identification:** Oluşturulma, son değiştirilme, son kullanma bilgilerini saklar.

5

File yaklaşımı



6

File yaklaşımı

Dosya işlemleri

- **Bir dosya soyut veri türüdür.**
- İşletim sistemi, dosya oluşturma, silme, okuma, yazma işlemleri için sistem çağrıları sağlayabilir.
 - **Dosya oluşturma:** Dosya için yeterli alan olmalı ve bir dizin içerisinde olmalıdır.
 - **Dosyaya yazma:** Dosya adına ve dosyaya yazılacak veri gereklidir. Sistem bir sonraki yazılacak konum için **write pointer** kullanır.
 - **Dosya okuma:** Dosyanın okunması sırasında sonraki konumu tutmak için **read pointer** kullanılır.
 - **Yeniden konumlandırma:** Dosya içerisinde konumlanmak için **seek** komutları kullanılır.
 - **Dosya silme:** Dizinde dosya adı aranır ve silme işlemiyle tüm alan serbest bırakılır.
 - **Dosyanın kesilmesi (truncate):** Dosyanın bir kısmı veya tüm veriler silinebilir.

7

File yaklaşımı

Dosya işlemleri

- **Dosya sonuna ekleme, dosya ismi değiştirme, dosya kopyalama, dosya taşıma gibi işlemler de yapılabilir.**
- Dosya işlemlerinin yapılabilmesi için öncelikle **open() sistem çağrısı ile dosyanın açılması gereklidir.**
- İşletim sistemi **tüm açık dosyalara ait bilgileri open-file table** ile tutar.
- Bazı işletim sistemlerinde dosya kullanımı bitince veya program kapatılınca **otomatik olarak dosya kapatılır.**
- **Open-file table ile her dosyayı açan process sayısı (open count) saklanır.**

8

File yaklaşımı

Dosya işlemleri

- **Açık durumdaki bir dosya** ile aşağıdaki bilgiler ilişkilendirilir:
 - **File pointer:** Sistem okuma ve yazma işlemi için şimdiki konum pointer'ı kullanır.
 - **File-open count:** Çok sayıda process aynı dosyaya erişebilir. Tüm process'ler dosyayı kapatıncaya kadar open-file table'da kaydı tutulur.
 - **Disk location:** Çoğu dosya işlemleri dosya içindeki verinin değişmesibi gerektirir. Dosyanın ilgili kısmı hafızaya alınarak değişiklik yapılır.
 - **Erişim hakları:** Process'ler bir dosyayı açarken erişim modu kullanır.
- Bazı işletim sistemleri dosyaları kilitleyerek **başka process'lerin erişimine kapatır.**
- **Shared lock (reader lock):** Birden fazla process eşzamanlı lock isteyebilir.
- **Exclusive lock (writer lock):** Aynı anda sadece bir process lock isteyebilir.

9

File yaklaşımı

Dosya türleri

- İşletim sisteminde dosya türünü belirlemek için yaygın kullanılan yöntemde, **dosya türü dosya adının bir kısmını oluşturur.**
- Bir dosya adı iki parçadan oluşur: **dosya adı** ve **uzantısı**.
`gunluk.docx` `sunucu.cpp` `maashesapla.xlsx`
- **Dosya türü** işletim sisteminin **dosya ile yapılacak işlemi gösterir (.exe, .com).**
- Kullanıcı **bilinen uzantının dışında uzantı belirleyebilir.**
- Farklı işletim sistemleri dosya türünü farklı şekilde belirler.
- **Mac OS X** dosya özelliklerinde **oluşturan uygulamayı belirtir.**
- **UNIX** bazı dosyaların başlangıcına **dosya türünü belirleyen numara (magic number) yazar.**

10

File yaklaşımı

Dosya türleri

file type	usual extension	function
executable	exe, com, bin or none	ready-to-run machine-language program
object	obj, o	compiled, machine language, not linked
source code	c, cc, java, perl, asm	source code in various languages
batch	bat, sh	commands to the command interpreter
markup	xml, html, tex	textual data, documents
word processor	xml, rtf, docx	various word-processor formats
library	lib, a, so, dll	libraries of routines for programmers
print or view	gif, pdf, jpg	ASCII or binary file in a format for printing or viewing
archive	rar, zip, tar	related files grouped into one file, sometimes compressed, for archiving or storage
multimedia	mpeg, mov, mp3, mp4, avi	binary file containing audio or A/V information

11

File yaklaşımı

Dosya yapısı

- Dosya türü, dosyanın iç yapısını göstermek için de kullanılabilir.
- Bazı dosyaların yapısının işletim sisteminin tanıyacağı şekilde olması zorunludur (.exe).
- İşletim sisteminin tanıdığı dosya türü arttıkça, işletim sistemi daha karmaşık hale gelir.
- Uygulamalar kendi dosya yapısını (**şifrelenmiş dosya**) oluşturabilir.
- UNIX, Windows ve diğer işletim sistemleri minimum dosya türünü tanır.
- Temel tanınan dosya türü çalıştırılabilir dosyadır (.exe).
- İletim sistemleri, diğer dosya türlerinin tanınmasını uygulama yazılımlarına bırakır.

12

File yaklaşımı

Dosyaların iç yapısı

- Disk sistemleri sektör (512 byte) olarak tanımlanan bloklardan oluşur.
- Mantıksal blok boyutu fiziksel blok boyutundan (sektör) farklı olabilir.
- UNIX, tüm dosyaları **byte stream'leri** olarak tanımlar. Her byte dosya başlangıcına göre offset adresine sahiptir.
- UNIX'te **mantıksal blok boyutu 1 byte, fiziksel blok boyutu 512 byte'dır.**
- Tüm dosyalar sıralı bloklardan oluşur.
- Disk alanı bloklar halinde yapılandırıldığından, **dosyaların son bloklarında her zaman boş kalan kısımlar olabilir.**
- Boyutu 1949 byte olan dosya 4 bloğa yazılır.
- $4 * 512 = 2048$ byte alan kullanılır. **99 byte boş kalır.**

13

Konular

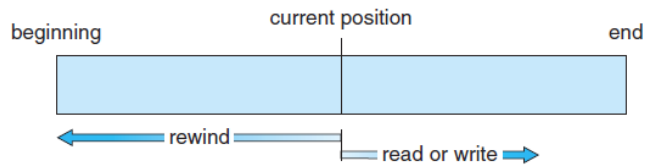
- File yaklaşımı
- Erişim yöntemleri
- Dizin ve disk yapısı
- Dosya sistemi oluşturma
- Dosya paylaşımı
- Koruma

14

Erişim yöntemleri

Sıralı erişim

- Dosya içerisindeki bilgiye sıralı erişilir (**sequential access**).
- Editör ve compiler yazılımları dosyaya sıralı erişirler.
- Read ve write işlemleri sıralı yapılır ve her işlemten sonra dosya pointer'ı otomatik olarak ilerletilir.
- Dosya üzerinde **bulunulan konum ileriye ve geriye doğru** atlanarak değiştirilebilir.



15

Erişim yöntemleri

Doğrudan erişim

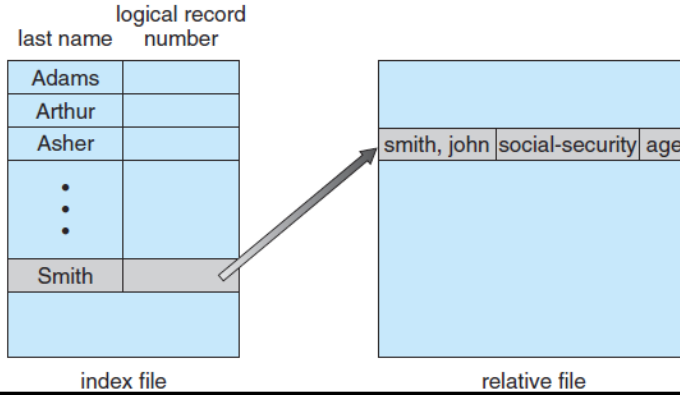
- Dosya içerisindeki kayıtlar sabit boyuttadır ve okuma/yazma işlemi doğrudan erişimle (**direct access**) hızlı bir şekilde yapılır.
- Doğrudan erişimde, dosya sıralanmış bloklar halinde görülür.
- Erişilmek istenen blok numarası kullanıcı tarafından işletim sistemine verilir. İlk kaydın sıra numarası 0 ile başlar.
- L blok boyutundaki bir dosyada N. kayda ulaşmak için, $L * N$ konumundaki kayda gidilir.

16

Erişim yöntemleri

Index kullanımı

- Bir dosya için **index dosyası** oluşturulabilir.
- Index, bir dosyadaki kayıtların pointer'ını içerir.
- Bir kaydı bulmak için **indeks'e bakılır**, varsa pointer ile konumlanılır.



17

Erişim yöntemleri

Index kullanımı - örnek

- Bir satış mağazasında **ürün kodu 10 basamak** ve **fiyat 6 basamak** olsun. Her kayıt **16 basamaktır (16 byte)**.
- Disk üzerinde **bloklar 1024 byte** ise her blokta **$1024 / 16 = 64$ kayıt saklanır**.
- Dosyada 120.000 kayıt varsa, **1.875 blok ($120.000 * 16 / 1024$)** gereklidir.
- Bir indeks her bloğun ilk elemanının ürün kodu bilgisini tutabilir.
- Index dosyası için **$1.875 * 10 = 18.750$ byte** gereklidir.
- Index'te binary arama yapılabilir.
- **$\log_2 1875 = 10,87 = 11$ maksimum karşılaştırma yapılır**.

18

Konular

- File yaklaşımı
- Erişim yöntemleri
- **Dizin ve disk yapısı**
- Dosya sistemi oluşturma
- Dosya paylaşımı
- Koruma

19

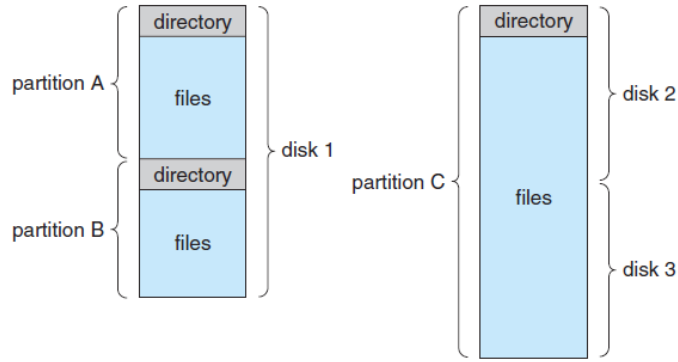
Dizin ve disk yapısı

- Her disk bölümü (**volume**) bir dosya sistemine sahiptir.
- Her **volume** dosya sistemine ilişkin bilgiyi tablo halinde saklar (**volume table of contents**).
- Her volüme, sahip olduğu tüm dosyalara ilişkin, adı, konumu, boyutu ve türü gibi bilgileri saklar.

20

Dizin ve disk yapısı

- Şekilde dosya sistemi organizasyonu görülmektedir.



- Bir volüme birden fazla fiziksel disk ile oluşturulabilir.

21

Dizin ve disk yapısı

Dizin yapısı

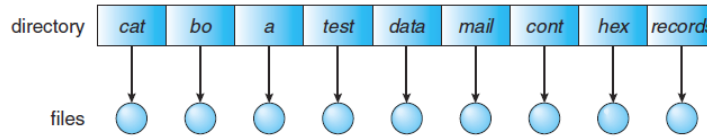
- Dosyalar bir dizinle ilişkilendirilerek saklanır.
- Dizin üzerinde aşağıdaki işlemler yapılabilir:
 - **Dosya arama:** Bir dizin içerisindeki dosyalar adına göre aranabilir.
 - **Dosya oluşturma:** Dizin içerisinde yeni dosya oluşturulabilir.
 - **Dosya silme:** Dizindeki dosya/lar silinebilir.
 - **Dizin listeleme:** Dizindeki dosyalar belirli kritere göre listelenebilir.
 - **Dosya adı değiştirme:** Dosya adında değişiklik yapılabilir.
 - **Dosya aktarımı:** Bir dizin içerisindeki **dosyalar** yedeklemek amacıyla **başka depolama birimine** aktarılabilir.

22

Dizin ve disk yapısı

Tek seviyeli dizin yapısı

- Tüm dosyalar aynı dizin içersindedir.
- Dosya sayısı arttıkça dosyaların bulunması ve tekil isimlendirilmesinde zorluklar ortaya çıkar.

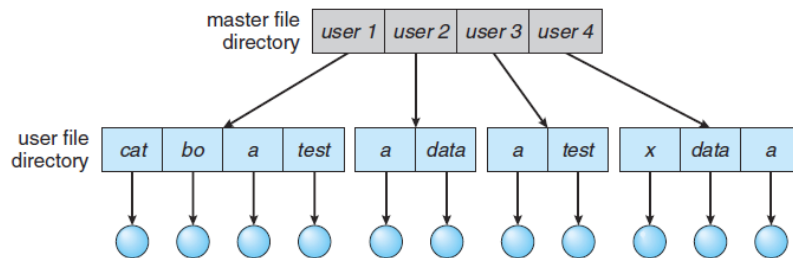


23

Dizin ve disk yapısı

İki seviyeli dizin yapısı

- Her kullanıcı kendisine ait dizine sahiptir.
- Bazı sistemlerde, kullanıcı kendisine ait dizindeki dosyalara erişebilir.
- Kullanıcılar birbirinden tamamen ayrıştırılabilir.



- Kullanıcı login olduğunda master file directory'de arama yapılır.
- Kullanıcı bulunursa user file directory'e yönlendirilir.

24

Dizin ve disk yapısı

İki seviyeli dizin yapısı

- Kullanıcı adı ile dosya adı path name olarak adlandırılır.
- Windows işletim sisteminde, `C:\kullaniciA\dosyalar\test.txt` şeklinde path verilir.
- UNIX ve Linux işletim sistemlerinde, `/c/kullaniciA/dosyalar/test.txt` şeklinde path verilir.
- VMS (Virtual Memory System)* işletim sisteminde, `C:[kullaniciA.dosyalar]test.txt;1` şeklinde path verilebilir. '1' versiyon numarasıdır.

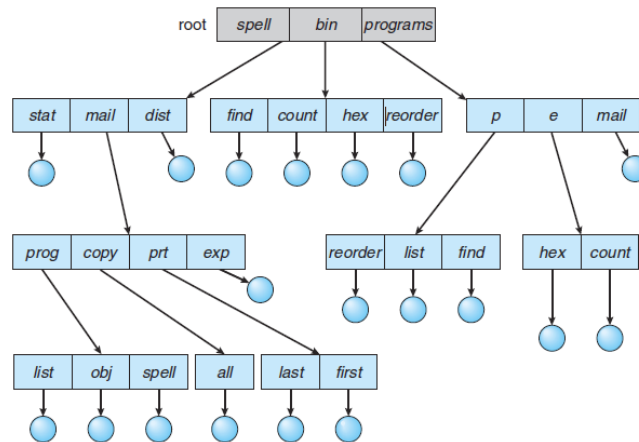
*VAX mikroişlemciler için geliştirilmiş işletim sistemidir.

25

Dizin ve disk yapısı

Ağaç dizin yapısı

- Doğal kullanıma en uygun yapı ağaç dizin yapısıdır.
- Kullanıcılar kendi dizinlerini yönetebilirler (yeni dizin oluşturma, silme,...)

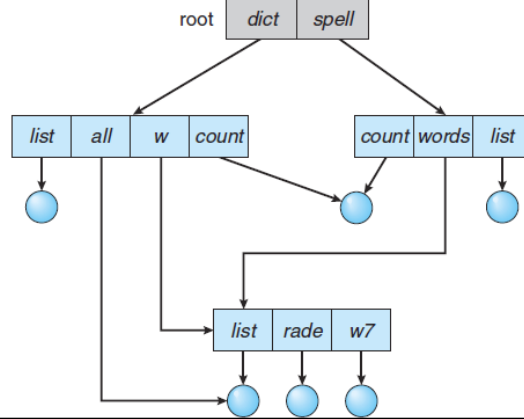


26

Dizin ve disk yapısı

Döngüsüz graf dizin yapısı

- Bir alt dizin veya dosya **kullanıcılar arasında paylaşılarak kullanılabilir**.
- **Alt dizin veya dosya silme işlemlerinin yönetimi zordur** (kimin sileceği belirlenmelidir, silinince diğerlerinin durumu belirlenmelidir).



27

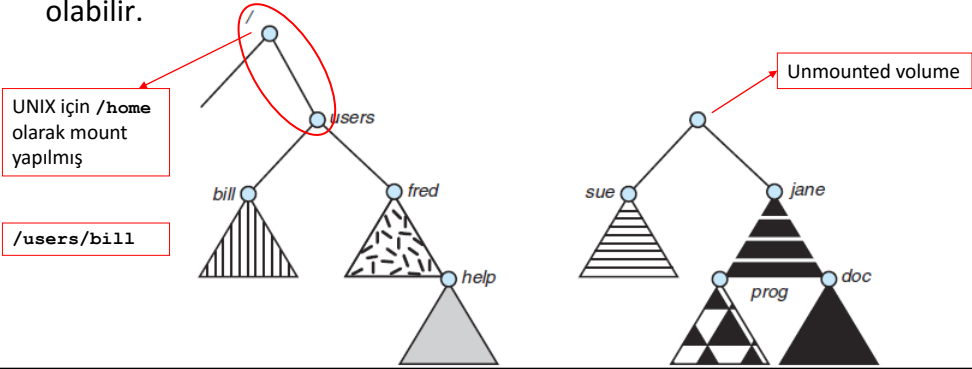
Konular

- File yaklaşımı
- Erişim yöntemleri
- Dizin ve disk yapısı
- **Dosya sistemi oluşturma**
- Dosya paylaşımı
- Koruma

28

Dosya sistemi oluřturma

- Dosya sistemi kullanılmadan önce **bir veya birden fazla disk üzerinde oluřturulmalıdır.**
- İřletim sistemine dosya sisteminin bulunduđu nokta (**mount point**) verilmelidir.
- Mount point, dosya sisteminin iliřkilendirildiđi bir cihaz veya dosya yapısı olabilir.



29

Konular

- File yaklařımı
- Eriřim yöntemleri
- Dizin ve disk yapısı
- Dosya sistemi oluřturma
- **Dosya paylařımı**
- Koruma

30

Dosya paylaşımı

Çok kullanıcıli sistemler

- Kullanıcılar arasında dosya paylaşımına sıklıkla ihtiyaç duyulur.
- Bir işletim sistemi çok kullanıcı ise, **dosya paylaşımı**, **dosya isimlendirme** ve **dosya koruma** en önemli konu haline gelir.
- Dosya paylaşımı ve koruması için daha fazla özellik oluşturulması gerekir.
- Çoğu sistem, **file owner** ve **group** yaklaşımlarının gelişmiş halini kullanır.
- File owner, dosya üzerinde daha fazla kontrol yetkisine sahiptir. (**Özellik değiştirebilir, erişim hakkı verebilir/alabilir.**)
- Group ise sadece dosyaya erişim paylaşım hakkı olan kullanıcı grubudur.
- **Group üyelerinin yapacağı işlemleri de file owner belirler.**
- Bir kullanıcı dosya üzerinde **işlem yapmak istediğinde yetkisi olup olmadığı kontrol edilir.**

31

Dosya paylaşımı

Uzak erişimli dosya sistemleri

- **Ağlar** farklı noktalarda bulunan **kullanıcıların dosyaları paylaşmasına izin verir.**
- Uzak noktada bulunan dosyalar **makinelere** arasında **transfer** edilerek paylaşılabilir (**ftp**).
- **Dağıtık dosya sistemlerinde** ise uzaktaki dosya kullanıcıların erişimine açılabilir.
- Günümüzde dosya paylaşımı için **World Wide Web (WWW)** giderek **yaygın kullanılmaya başlanmıştır.**
- Bir browser uzaktaki dosyaya **erişim hakkı alır ve üzerinde işlem yapar.**
- ftp **anonim** veya **kimliği doğrulanmış** erişime izin verir.
- **Anonim erişimde account (kullanıcı adı ve şifre) gerekmez.**

32

Konular

- File yaklaşımı
- Erişim yöntemleri
- Dizin ve disk yapısı
- Dosya sistemi oluşturma
- Dosya paylaşımı
- **Koruma**

33

Koruma

Erişim türleri

- Bilginin **fiziksel bozulmalara** ve **yetkisiz erişimlere** karşı korunması (**protection**) gereklidir.
- **Reliability** (güvenilirlik), dosyaların birden fazla kopyasının alınmasıyla sağlanır. **Sorun olduğunda yedek geri yüklenir.**
- Başka kullanıcıların erişimine izin vermeyen sistemlerde protection gerekli değildir.
- Koruma, **erişimi yasaklamak** veya **kısıtlamakla** sağlanır.
- **Erişimi kısıtlamak**, başka kullanıcıların yapabileceği işlemlerin kısıtlanması şeklinde gerçekleştirilebilir.

34

Koruma

Erişim türleri

- Dosyalar üzerinde **çok sayıda işlem kısıtlanabilir**:
 - **Read**: Dosyadan okuma
 - **Write**: Dosyaya yazma
 - **Execute**: Dosyayı hafızaya yükleyip çalıştırma
 - **Append**: Dosya sonuna yeni bilgi yazma
 - **Delete**: Dosyayı silme
 - **List**: Dosya ismini veya özelliklerini listeleme
- Bunların dışında, **isim değiştirme, kopyalama, taşıma, düzenleme** gibi işlemler de kısıtlanabilir.

35

Koruma

Erişim denetimi

- En yaygın kullanılan koruma yöntemi, **kullanıcı kimliklerine bağlı erişim yapmaktır**.
- **Her dosya için** erişim yapacak kullanıcılar ile erişim türlerini içeren **erişim listesi (access-control list) oluşturulabilir**.
- **Her dosya için kullanıcı bazında liste oluşturmak ve yönetmek zordur**.
- Her dosya için **üç grup kullanıcı oluşturulur**:
 - **Owner**: Dosyayı oluşturan kullanıcıdır.
 - **Group**: Dosyayı paylaşan ve benzer erişim ihtiyaçlarına sahip olan kullanıcılarıdır.
 - **Universe**: Sistemdeki diğer tüm kullanıcılarıdır.

36

Koruma

Erişim denetimi - UNIX

- UNIX, üç grup altında her dosya için erişim denetimi yapar.

-rw-rw-r--	1	pbg	staff	31200	Sep 3 08:30	intro.ps
drwx-----	5	pbg	staff	512	Jul 8 09:33	private/
drwxrwxr-x	2	pbg	staff	512	Jul 8 09:35	doc/
drwxrwx---	2	jwg	student	512	Aug 3 14:13	student-proj/
-rw-r--r--	1	pbg	staff	9423	Feb 24 2012	program.c
-rwxr-xr-x	1	pbg	staff	20471	Feb 24 2012	program
drwx--x--x	4	tag	faculty	512	Jul 31 10:31	lib/
drwx-----	3	pbg	staff	1024	Aug 29 06:52	mail/
drwxrwxrwx	3	pbg	staff	512	Jul 8 09:35	test/

- **d** = altdizin, **r** = read, **w** = write, **x** = execute işlemini tanımlar.
- İlk üç rwx **owner** için tanımlanmıştır.
- İkinci rwx **group kullanıcılar** için tanımlanmıştır.
- Üçüncü rwx **diğer kullanıcılar** için tanımlanmıştır.

37

Koruma

Erişim denetimi - Windows

- Windows, kullanıcı grupları oluşturmaya izin verir.
- Şekilde Administrator kullanıcı grubu OS_L10FileSystemInterface.ppt dosyasına erişim hakkına sahiptir.

