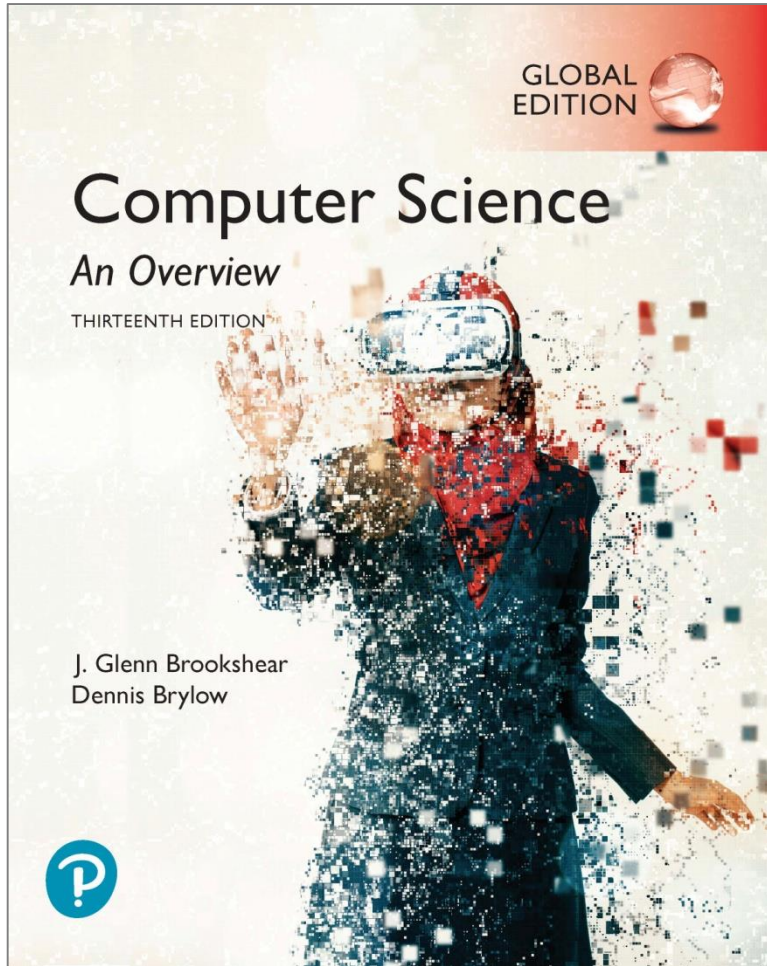


# Bilgisayar Bilimine Giriş

13<sup>th</sup> Baskı, Global Edition



## Bölüm 3

### İşletim Sistemleri

# Bölüm 3 : İşletim Sistemleri

- 3.1 İşletim Sistemlerinin Tarihi
- 3.2 İşletim Sistemi Mimarisi
- 3.3 Makinenin Aktivitelerini Yönetmek
- 3.4 İşlemler Arasındaki Yarışı Yönetmek
- 3.5 Güvenlik

# İşletim Sistemi Örnekleri

- Windows
- UNIX
- Mac OS
- Solaris (Sun/Oracle makineleri)
- Linux

# Akıllı Telefon İşletim Sistemleri

- Apple iOS
- Windows Phone
- BlackBerry OS
- Nokia Symbian OS
- Google Android

# İşletim Sisteminin İşlevleri

- Bilgisayarın yaptığı işleri denetler
- Dosyaları depolar ve kurtarır
- Programların çalışması için gereken girdi için kullanıcıya arayüz sağlar
- Programların çalışmasını yönetir

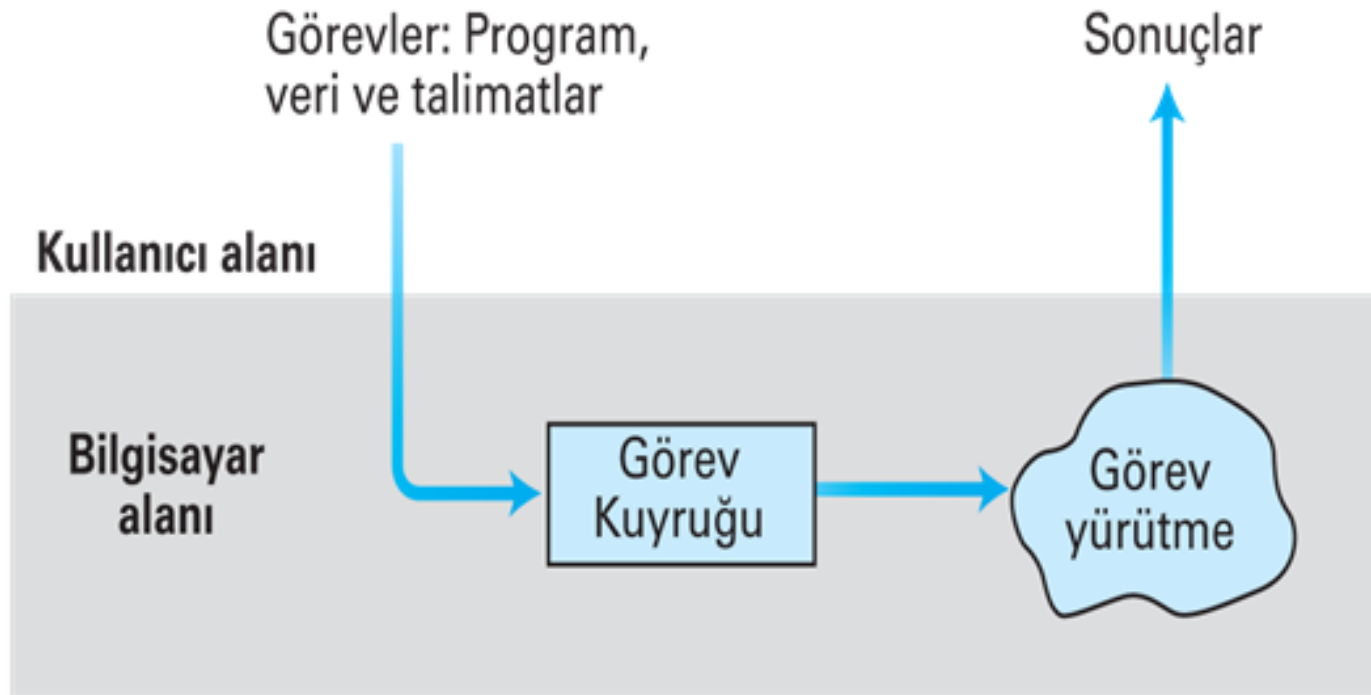
## 3.1 İşletim Sistemlerinin Tarihi

- Her bir programa “Görev” denir
- İlk bilgisayarlarda bir «Görev» çalıştırmak için oldukça uzun ve karmaşık işlemler gerekiyordu.
- Her “Görev” kendi kurulumuna ihtiyaç duymaktaydı.
- İşletim sistemleri Görevler arasında geçiş sağlamak ve program kurulumlarını kolaylaştıracak sistemler olarak ortaya çıktı

## 3.1 İşletim Sistemlerinin Tarihi

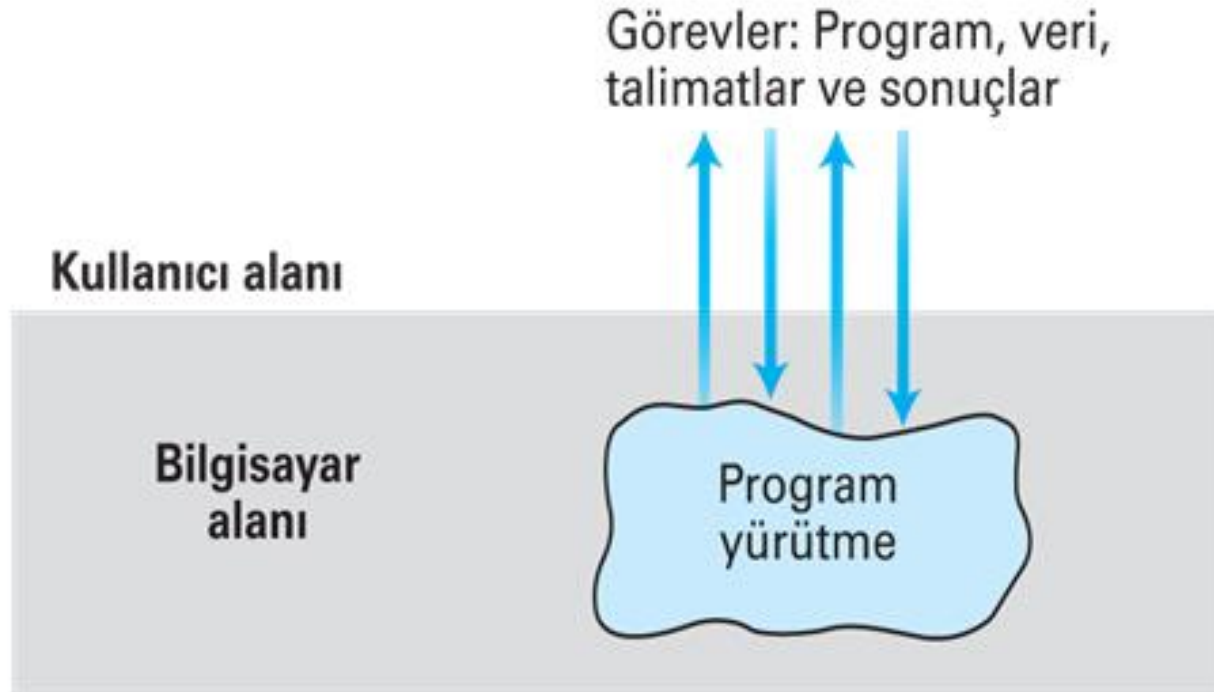
- Toplu işlem(iş sırası)
- Etkileşimli işlem (gerçek zamanlı)
- Zaman-paylaşımı (bir makine,birden fazla kullanıcı)
- Çok görevlilik (bir kullanıcı , birden çok işlem)
- Çok işlemcili makineler (yük dengeleme)
- Gömülü sistemler(belli aygıtlarda)

## Şekil 3.1 Toplu işlem





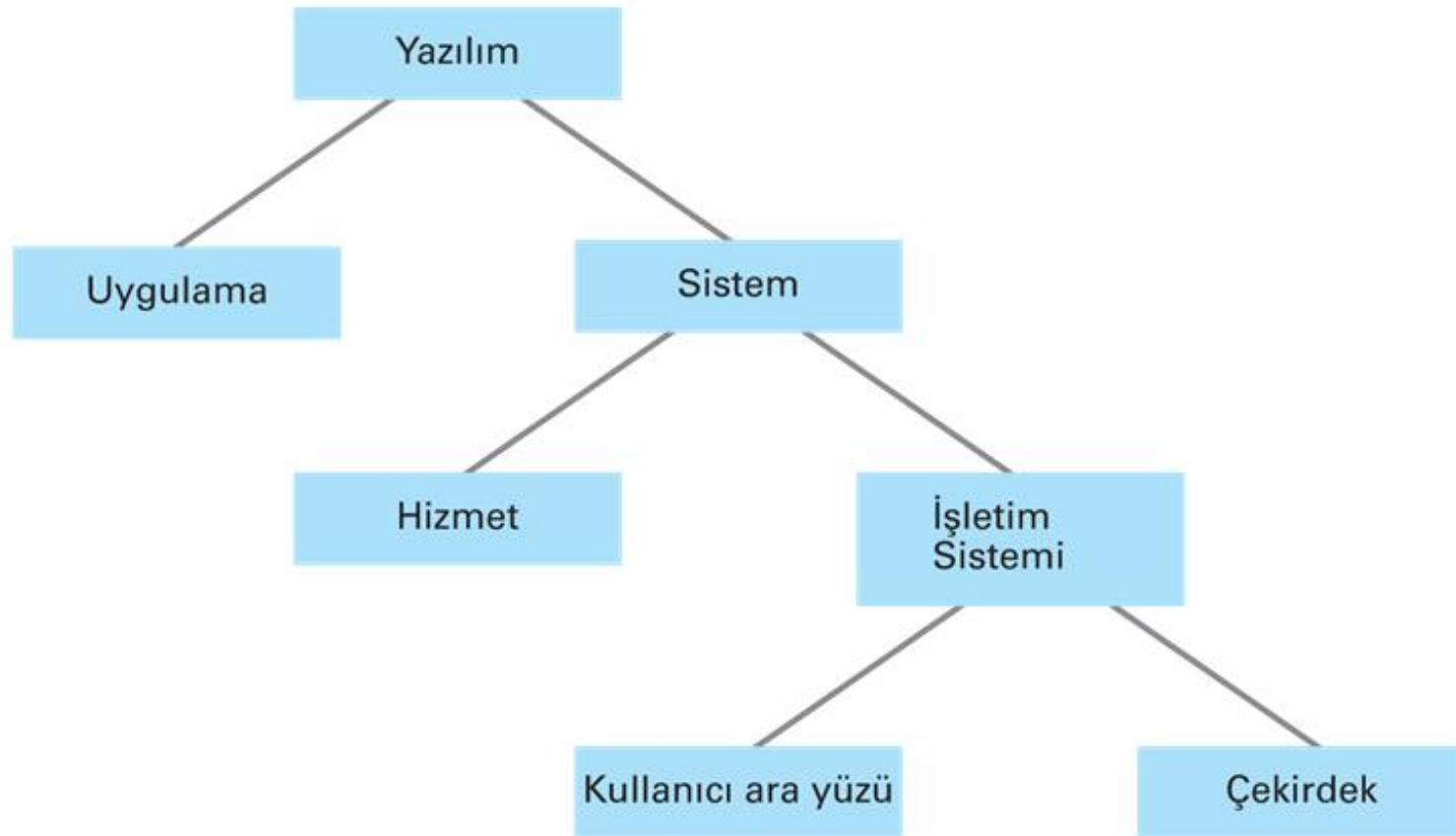
## Şekil 3.2 Etkileşimli İşlem



## 3.2 İşletim Sistemi Mimarisi

- Uygulama yazılımı
  - Kullanıcı için belirli özel komutları yerine getirir (ofis programları, oyunlar,yazılım geliştirme)
- Sistem yazılımı
  - Uygulama yazılımına altyapı sağlar
  - İşletim sisteminden ve işletim sistemine destek yazılımlarından oluşur

## Şekil 3.3 Yazılım Sınıflandırma



# İşletim Sistemi Bileşenleri

- **Kullanıcı arayüzü:** Kullanıcının muhatap olacağı bölümdür
  - Metin tabanlı (Kabuk)
  - Grafik kullanıcı arayüzü (GUI)
- **Kernel:** Gerekli temel fonksiyonları çalıştırır
  - Dosya yöneticisi
  - Aygıt sürücüler
  - Bellek yöneticisi
  - Zamanlayıcı ve programlayıcı

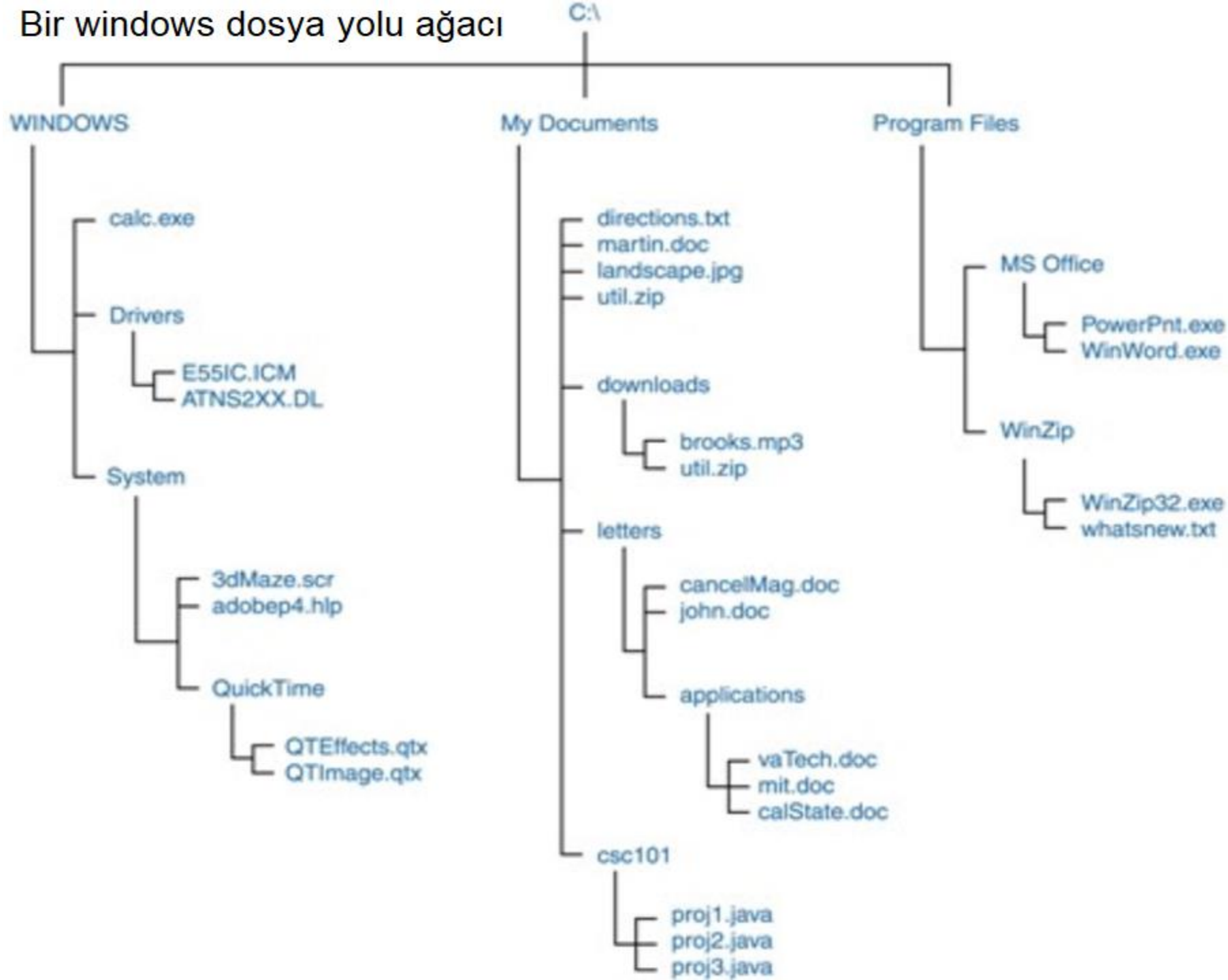
## Şekil 3.4 Kullanıcı arayüzü, kullanıcı ve sistemin kernel'i arasında bir aracıdır



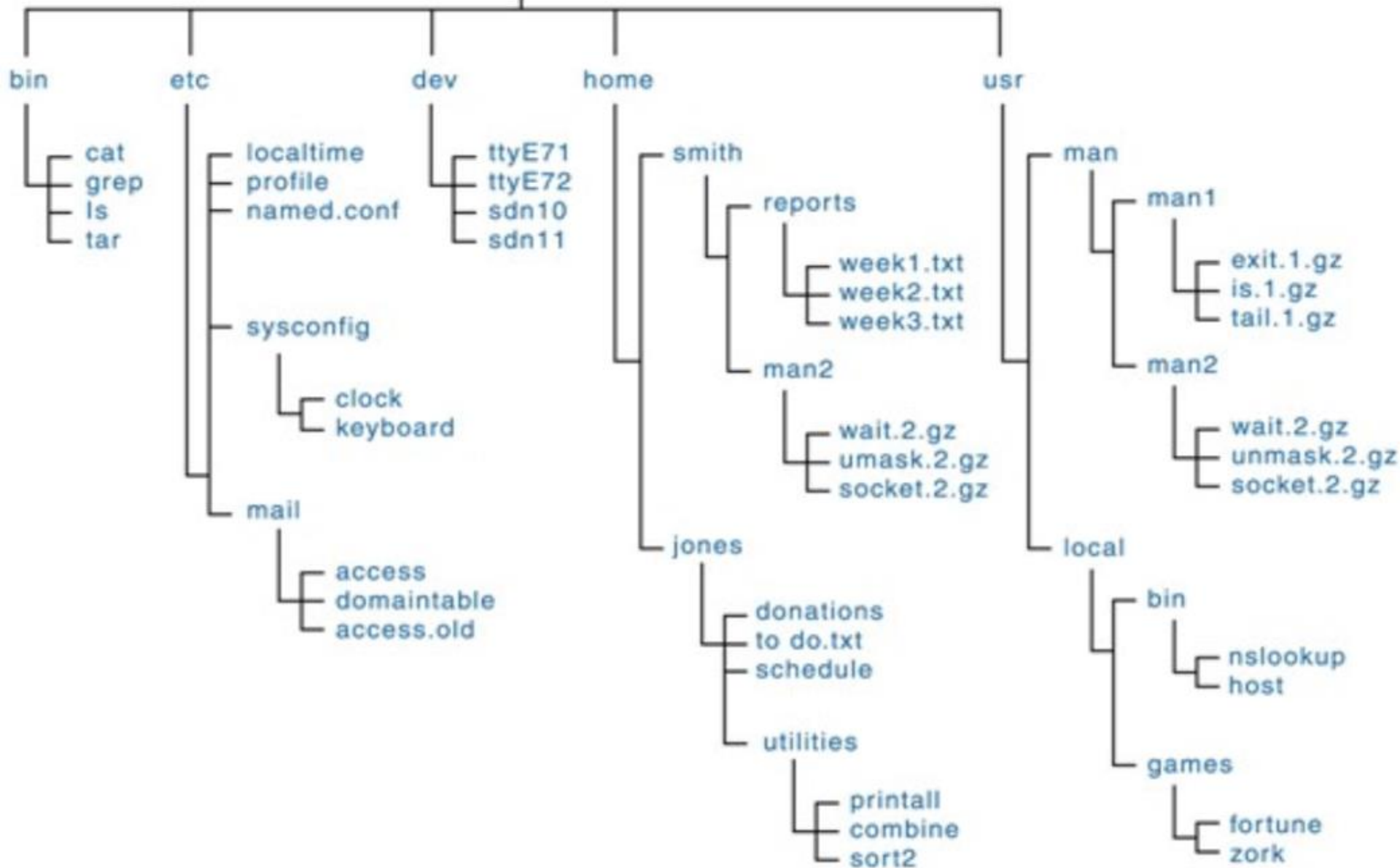
# Dosya yöneticisi

- **Dizin(Klasör):** Kullanıcı tarafından oluşturulmuş dosya paketi ve diğer yollar
- **Dizin Yolu:** Yolların içinde yolların dizileridir

# Bir windows dosya yolu ağacı



## Bir UNIX dosya yolu ağacı





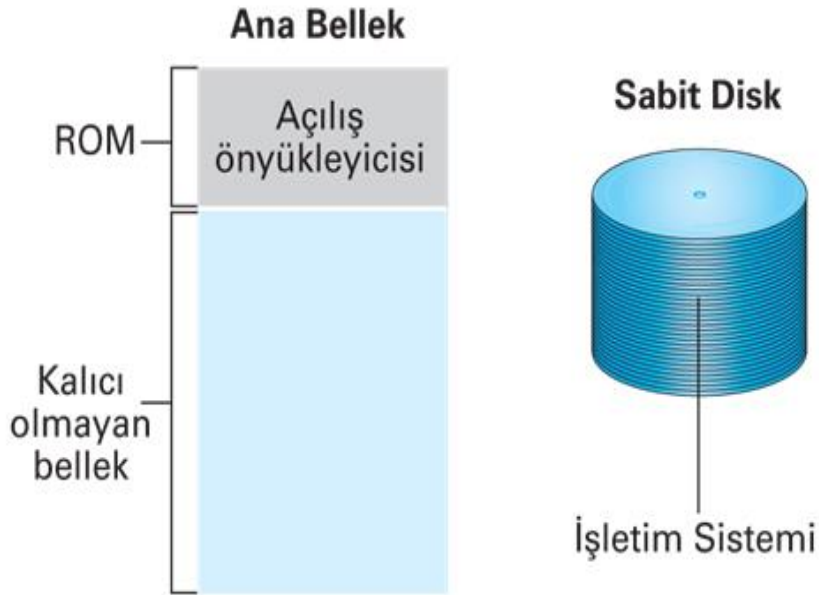
# Bellek Yöneticisi

- Ana bellekteki alanı paylaşırır
- Bellek yöneticisini işi toplam bellek ihtiyacının bilgisayarda bulunan bellek alanından fazla olması durumunda daha da karmaşıklaşır. Bu durumda, bellek yöneticisi programları veya veriyi ana bellekle sabit disk arasında taşıyarak fazladan bellek alanı varmış izlenimi yaratır. Bu tekniğe sayfalama (paging) adı verilir

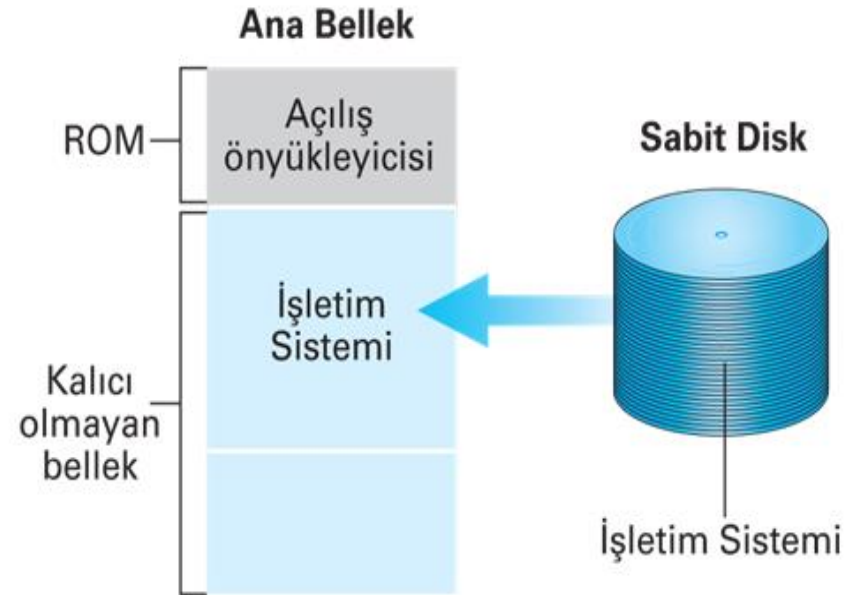
# Başlatmak (Booting)

- **Önyükleyici:** ROM'daki program (Bellenim örneği)
  - Güç verildiği zaman CPU ile çalışır
  - İşletim sistemini yığın depolamadan ana belleğe aktarır
  - İşletim sisteminin çalışmaya başlamasını sağlar

## Şekil 3.5 Başlama işlemi



**Adım 1:** Bilgisayar başlangıçta bellekte bulunan açılış önyükleyicisini yürüterek başlar. İşletim sistemi sabit diskte dir.



**Adım 2:** Açılış önyükleyicisi işletim sisteminin ana belleğe taşınmasını sağlar ve sonra kontrolü işletim sistemine devreder.

## 3.3 Bilgisayarın aktivitelerini koordine etme

Bir işletim sistemi;

- uygulama yazılımlarının,
- faydalı yazılımların ve
- işletim sisteminin içindeki birimlerin kendilerini yönetir

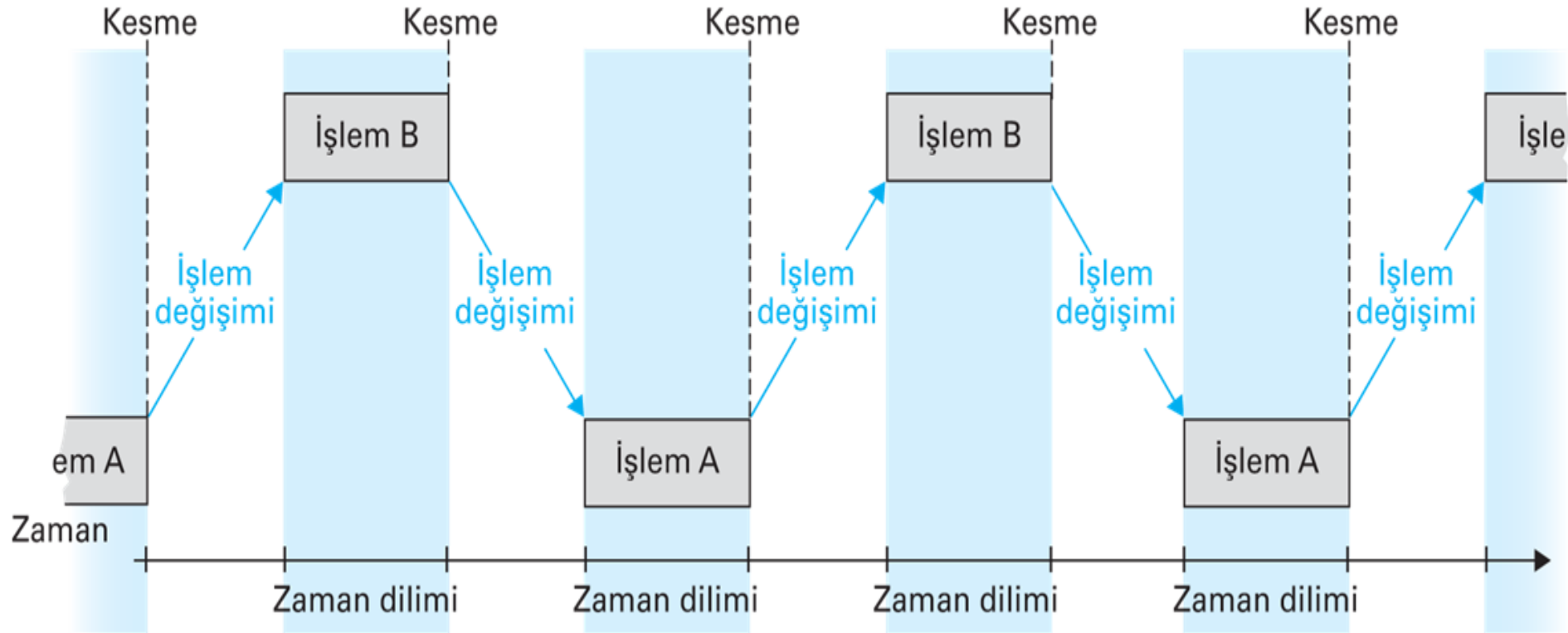
# İşlem Kavramı

- **İşlem:** Bir programı çalıştırma aktivitesi
- **İşlem durumu:** Aktivitenin mevcut durumu
  - Program sayacı
  - Genel amaçlı yazmaçlar
  - Ana belleğin ilgili kısımları

# İşlem yönetimi

- **Zamanlayıcı** : İşlem tablosuna yeni işlemleri ekler ve tamamlanmış işlemleri tablodan kaldırır
- **Sevk Edici**: İşlem tablosundaki işlemlerin zaman dilimlerine paylaştırılmasını kontrol eder
  - Bir zaman diliminin sonu bir kesme işaretiyle işaretlenir.

# Şekil 3.6 İşlem A ve İşlem B arasındaki Çoklu Programlama



## 3.4 İşlemler Arasındaki Yarışı Yönetmek

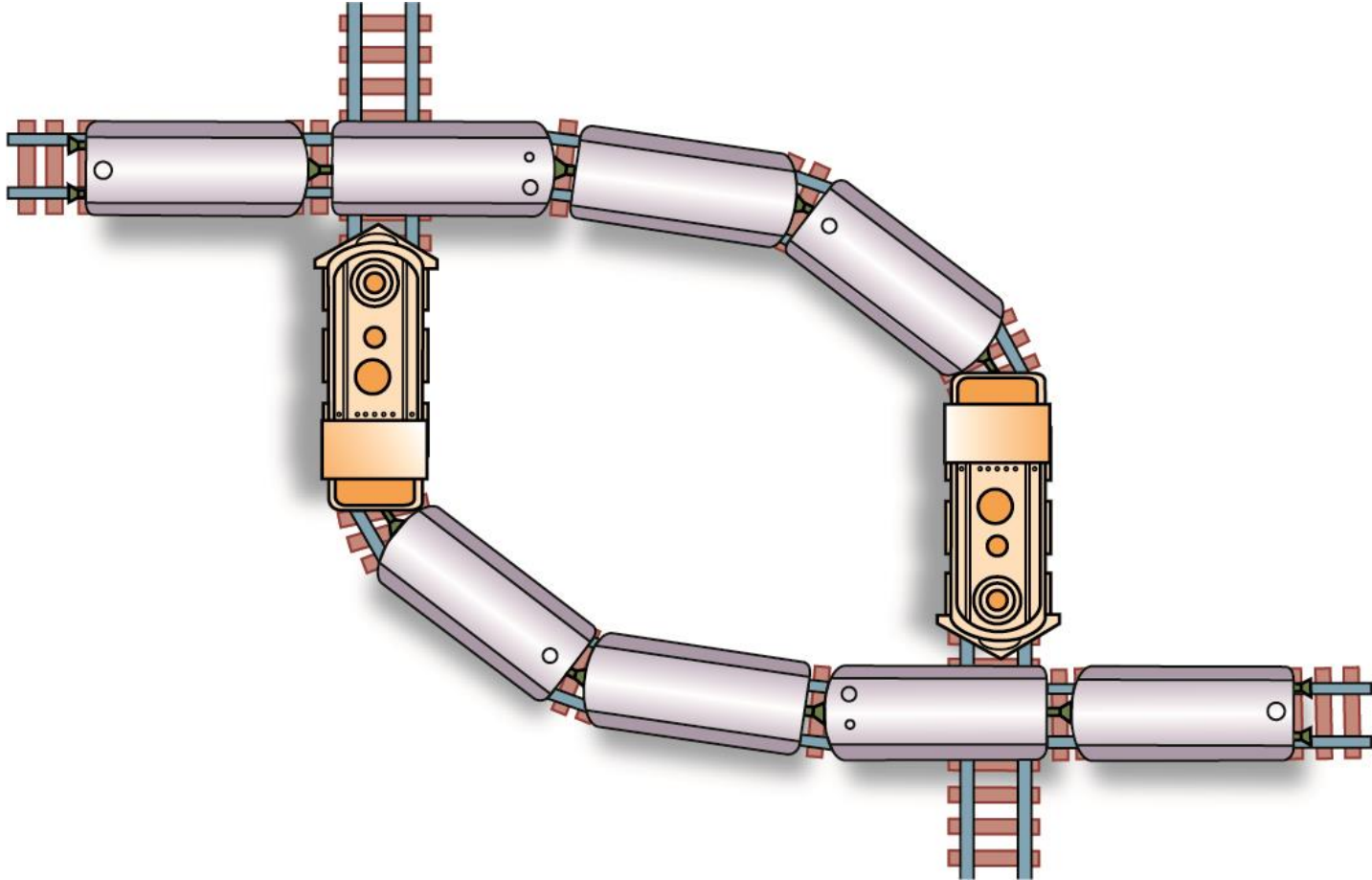
- **Semafor:** Bir “kontrol bayrağı”
- **Kritik bölge:** Tek seferde sadece bir işlem yapılması gereken talimat grubu



# Kilitlenme (Deadlock)

- Kilitlenme: iki yada daha fazla işlemin birbirini tarafından , her birinin ,diğerine pay edilmiş bilgiyi beklediği için bloklanmasıdır .
- Mesela , bir işlem bilgisayarın yazıcısını kullanıyor fakat CD oynatıcısına erişim için bekliyor; bir diğer işlem de CD oynatıcısını kullanıyor fakat yazıcıya erişim için bekliyor

## Şekil 3.7 Paylaşılmayan bir demir yolu bölümü için yarışma sonucu oluşan kitlenme



## 3.5 Güvenlik

- Dış kaynaklı saldırılar
  - Sorunlar
    - Güvenli olmayan şifreler
    - Yazılım kokuşma(sniffing software)

# Güvenlik

- İç kaynaklı saldırılar
  - Sorun: Bir işlem bellekte kendine ayrılmış bölge dışında erişim sağlar