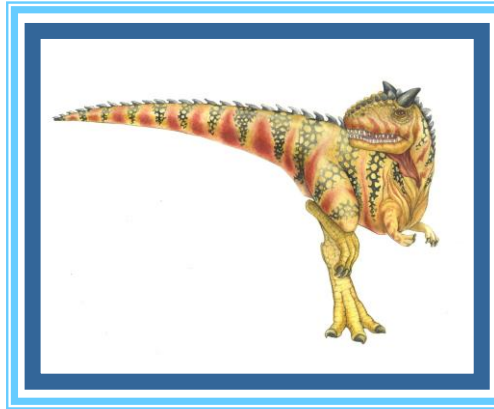


# Bölüm 1: Giriş





# İçindekiler

---

- İşletim Sistemleri Ne Yapar?
- Bilgisayar Sistemi Organizasyonu
- Bilgisayar Sistemi Mimarisi
- İşletim Sistemi Operasyonları
- Kaynak Yönetimi
- Güvenlik ve Koruma
- Sanallaştırma
- Dağıtık Sistemler
- Çekirdek(Kernel) Veri Yapıları
- Bilgi İşlem Ortamları
- Ücretisiz/Özgür ve Açık Kaynak İşletim Sistemleri





# Amaçlar

---

- Bilgisayar sisteminin genel organizasyonunu ve kesmelerin rolünü öğrenmek
- Modern, çok işlemcili bir bilgisayar sistemindeki bileşenleri açıklamak
- Kullanıcı modundan çekirdek moduna geçişi göstermek
- İşletim sistemlerinin çeşitli bilgi işlem ortamlarında nasıl kullanıldığını tartışmak
- Ücretsiz ve açık kaynaklı işletim sistemlerine örnekler sağlamak





# İşletim Sistemi Terimi Ne Anlama Geliyor?

---

- Bir işletim sistemi kullanıcı ve donanım arasındaki eksik kısmı dolduran bir ara katmandır
  - Program
  - Donanım
  - Derleyici
  - Vs.



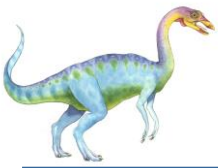


# İşletim Sistemi Nedir?

---

- Bilgisayar kullanıcısı ile bilgisayar donanımı arasında aracılık yapan program
- İşletim sistemi hedefleri:
  - Kullanıcı programlarını yürütme ve kullanıcı sorunlarını çözmeyi kolaylaştırma
  - Bilgisayar sistemini kullanımına uygun hale getirme
  - Bilgisayar donanımını verimli bir şekilde kullanma



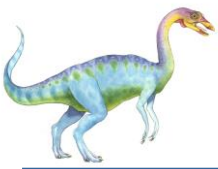


# Bilgisayar Sistemi Yapısı

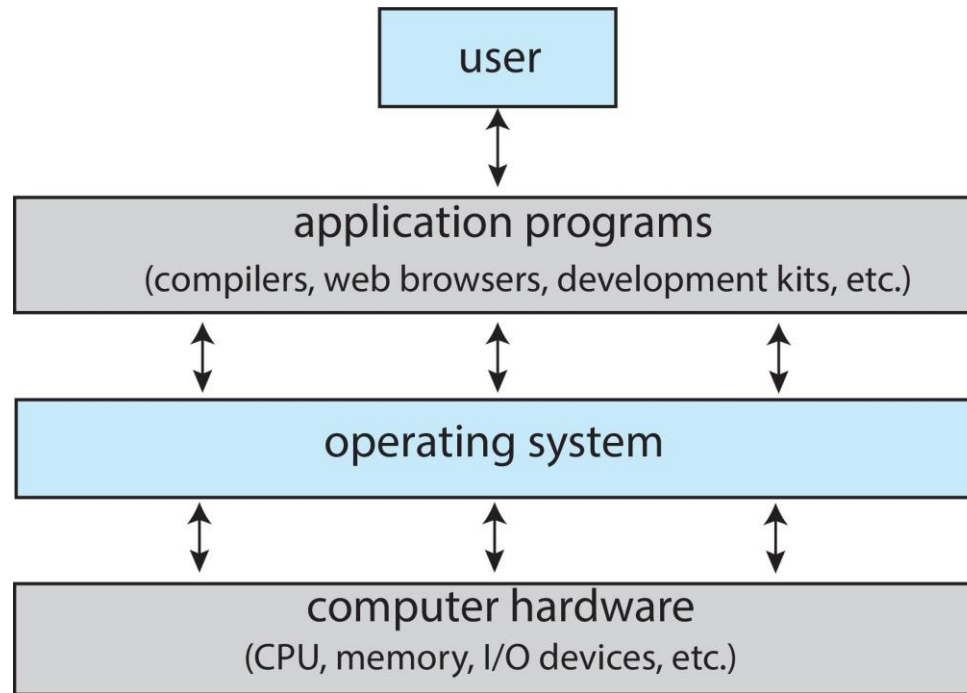
---

- Bilgisayar sistemi dört bileşene ayrılabilir:
  - Donanım – temel bilgi işlem kaynakları sağlar
    - ▶ CPU, bellek, I/O aygıtları
  - İşletim Sistemi
    - ▶ Çeşitli uygulamalar ve kullanıcılar arasında donanım kullanımını kontrol ve koordine eder
  - Uygulama programları – kullanıcıların bilgi işlem sorunlarını çözmek için sistem kaynaklarının nasıl kullanıldığını tanımlar
    - ▶ Kelime işlemciler, derleyiciler, web tarayıcıları, veritabanı sistemleri, video oyunları
  - Kullanıcı
    - ▶ İnsanlar, makineler, diğer bilgisayarlar





# Bilgisayar Bileşenlerinin Soyut Görünümü





# İşletim Sistemleri Ne Yapar?

---

- Bakış açısına bağlıdır
- Kullanıcılar kolaylık, **kullanım kolaylığı** ve **iyi performans** ister
  - **Kaynak Kullanımı** ile ilgilenmezler
- Ancak **anabilgisayar** veya **mini bilgisayar** gibi paylaşılan bilgisayarlar da tüm kullanıcıları mutlu etmelidir
  - İşletim sistemi donanımdan verimli bir şekilde yararlanmayı ve kullanıcı programlarının yürütülmesini yöneten bir **kaynak tahsis edici (allocator)** ve **kontrol programıdır**.



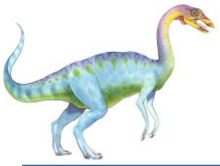




# İşletim Sistemleri Ne Yapar (devam)

- **İş istasyonları** gibi sistemlerin kullanıcıları ayrılmış kaynaklara sahiptir, ancak **sunucunun** paylaşılan kaynaklarını sıkça kullanırlar.
- Akıllı telefonlar ve tabletler gibi mobil cihazların kaynakları zayıftır, kullanılabilirlik ve pil ömrü için optimize edilmiştir
  - Dokunmatik ekranlar, ses tanıma gibi mobil kullanıcı arayüzleri
- Bazı bilgisayarlarda, aygıtlar içinde ve otomobillerdeki gömülü bilgisayarlar gibi çok az kullanıcı arabirimi vardır veya hiç yoktur
  - Öncelikle kullanıcı müdahalesi olmadan çalışır





# OS Terimi Birçok Rolü İçerir

---

- Çok sayıda tasarım ve işletim sistemi kullanımının olması ana sebeptir
- Gemiler, uzay araçları, oyun makineleri, TV'ler ve endüstriyel kontrol sistemleri içinde bulunur
- Askeriye için sabit kullanımlı bilgisayarlar için geliştirilmiş ancak kaynak yönetimine ve program kontrolüne ihtiyaç duyarak daha genel amaçlı bir hale gelmiştir





# İşletim Sistemi Tanımı

- Evrensel olarak kabul edilen ortak bir tanım yok
- "Bir işletim sistemi sipariş ettiğinizde satıcı her şeyi gönderir" iyi bir yaklaşımdır
  - Ancak teknoloji hızla değişir.
- "Bilgisayarda her zaman çalışan tek bir program" **çekirdek** olarak adlandırılır ve işletim sisteminin bir parçasıdır
- Geriye kalan her şey:
  - Ya bir **sistem programıdır** (işletim sistemiyle birlikte, ancak çekirdeğin bir parçası değil) veya
  - Ya da bir **uygulama programıdır**, işletim sistemiyle ilişkili olmayan tüm programlar
- İster genel amaçlı ister mobil bilgi işleme için olsun günümüzün işletim sistemleri bir **ara yazılımdır** – veritabanları, multimedya, grafikler gibi uygulama geliştiricilerine ilave hizmetler sağlayan bir yazılım kümesi





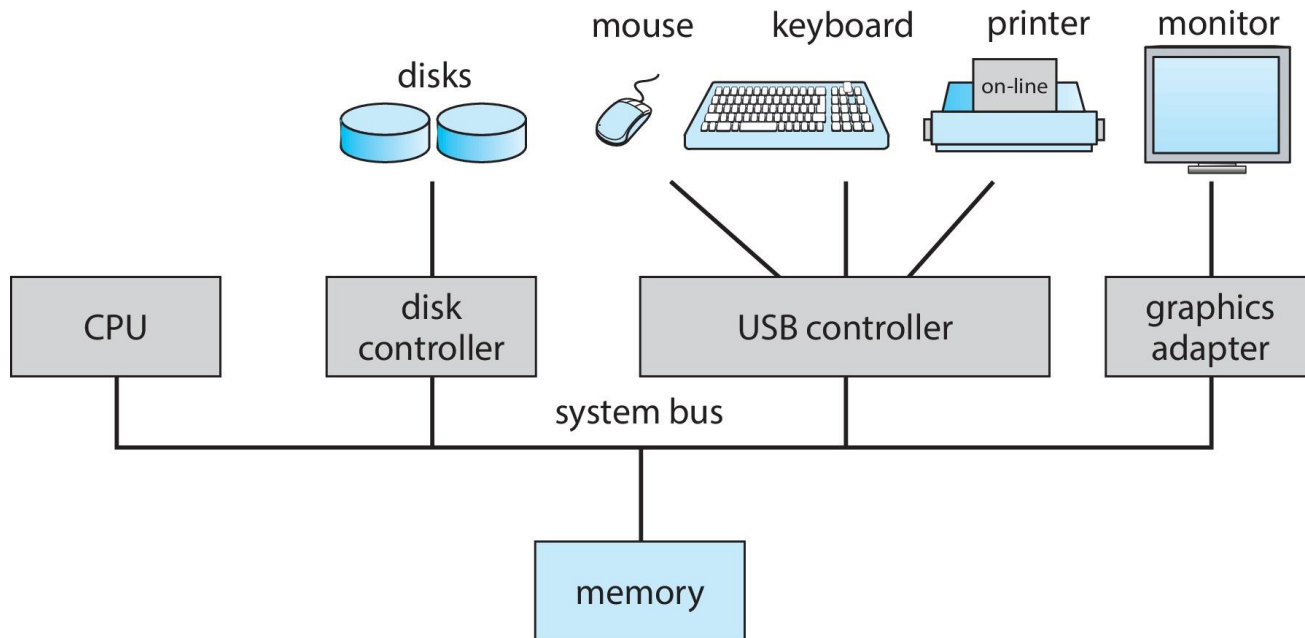
# Bilgisayar Sistemi Yapısına Genel Bakış

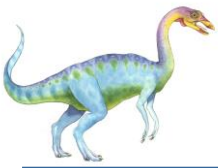




# Bilgisayar Sistemi Organizasyonu

- Bilgisayar sistemi çalışması
  - Bir veya daha fazla CPU, cihaz denetleyicileri ortak **veriyolu-bus** ile paylaşılan belleğe erişim sağlar
  - CPU'ların ve bellek için yarışan cihazların eşzamanlı yürütülmesi





# Bilgisayar Sistemi Çalışması

- I/O aygıtları ve CPU aynı anda çalışabilir
- Her cihaz denetleyicisi belirli bir cihaz türünden sorumludur
- Her aygıt denetleyicisinin yerel tamponu vardır
- Her aygıt denetleyicisinin **aygıt sürücüsü** adında bir sistem yazılımı vardır
- CPU verileri ana bellekten yerel tamponlara/arabelleklere veya tam tersi yönde taşır
- I/O aygıttan denetleyicinin yerel arabelleğine doğru olur
- Aygıt denetleyicisi CPU'ya, işlemini tamamladığını bir **kesme-interrupt** ile bildirir





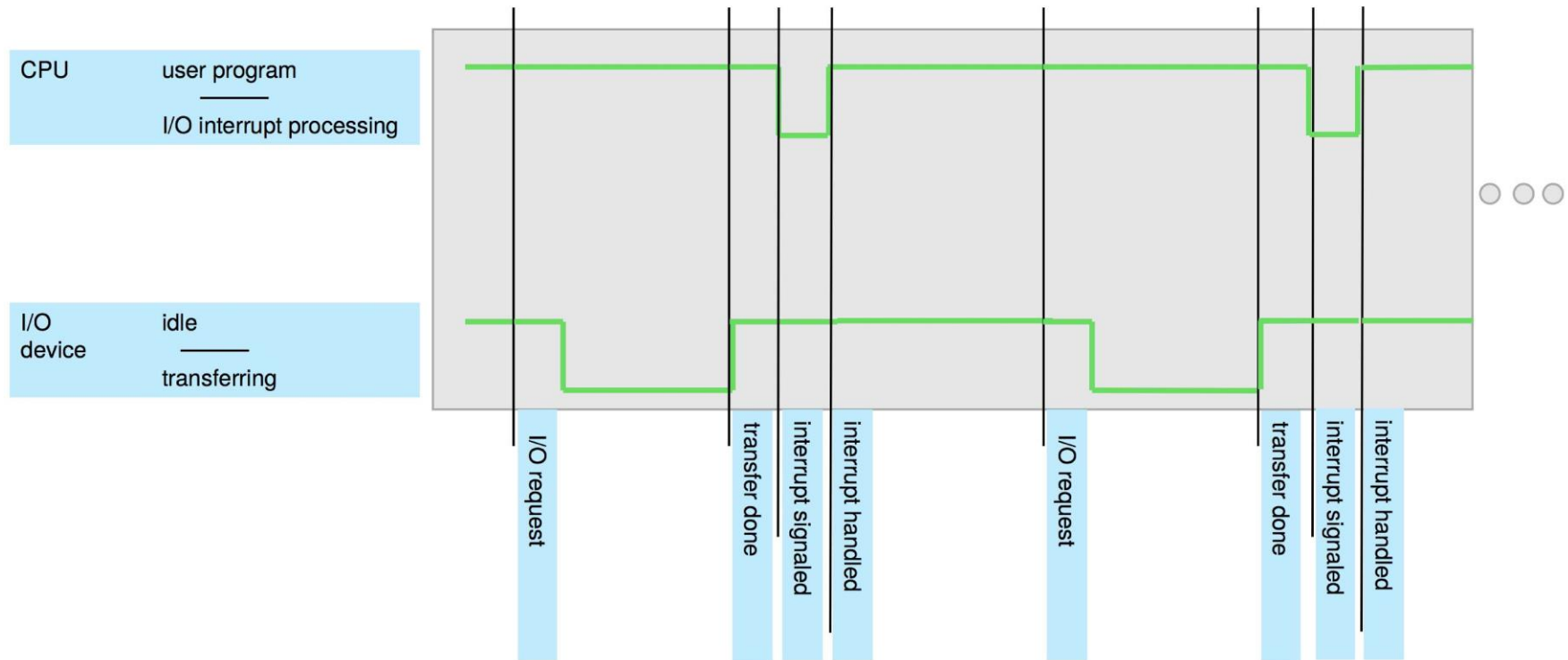
# Kesmelerin Ortak İşlevleri

- Kesme, denetimi kesme hizmeti yordamına iletir. Tüm servis yordamlarının adreslerini içeren bir **kesme vektörü** kullanılır.
- Kesme mimarisi kesintiye uğrayan yönergenin adresini kaydetmelidir
- A **hata bildirimi** veya **istisna** bir hatanın veya kullanıcı isteğinin neden olduğu yazılım tarafından oluşturulan bir kesmedir
- İşletim sistemi **kesme-güdümlüdür**

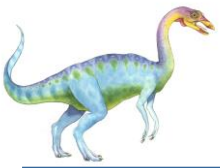




# Kesme Zaman Çizelgesi





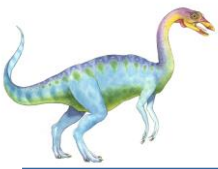


# Kesme Yönetimi

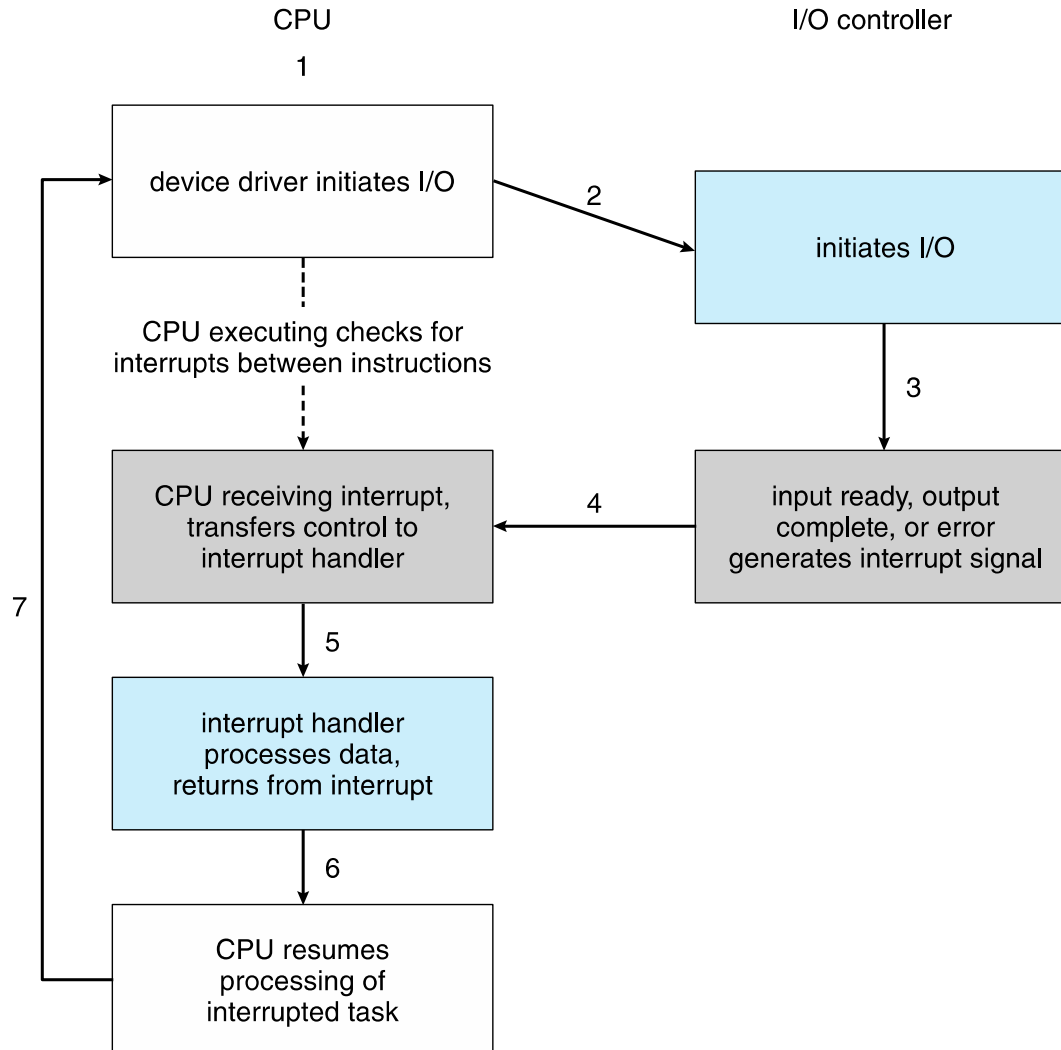
---

- İşletim sistemi, kaydedicileri- registers ve program sayacını yedekleyerek CPU'nun durumunu kaydeder.
- Hangi kesme türünün oluştuğunu belirler
- Ayrı kod kesimleri, her kesme türü için hangi eylemin yapılması gerektiğini belirler





# Kesme sürücüsü I/O Döngüsü



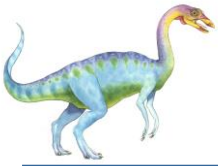


# I/O Yapısı

---

- I/O yönetimi için iki yöntem
  - I/O başlatıldıktan sonra, yalnızca I/O tamamlanırsa kontrol kullanıcı programına geri döner
  - I/O başladıktan sonra, I/O nun tamamlanmasını beklemeden kontrol kullanıcı programına geri döner





# I/O Yapısı (devam)

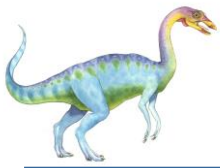
- I/O başlatıldıktan sonra, yalnızca I/O tamamlanırsa kontrol kullanıcı programına geri döner
  - Wait talimatı bir sonraki kesmeye kadar CPU'yu boşa bekletir
  - Wait döngüsü (bellek erişimi için yarış)
  - Çoğu bir I/O isteği aynı anda olağanüstüdür, eşzamanlı I/Ç işlemi yoktur
- I/O başladıktan sonra, I/O nun tamamlanmasını beklemeden kontrol kullanıcı programına geri döner
  - **Sistem çağırısı** – kullanıcının I/O tamamlanmasını beklemesi için OS ye yapılan istek
  - **Aygıt durum tablosu** her bir I/O aygıtın türünü, adresini ve durumunu içerir
  - OS, aygıt durumunu belirlemek ve tablo girdisini kesmeyi içerecek şekilde değiştirmek için I/O aygıt tablosunu günceller.





# Depolama Yapısı





# Depolama Yapısı

---

- Ana bellek – yalnızca CPU'nun doğrudan erişebileceği büyük depolama ortamı
  - Genellikle **uçucu**
  - Genellikle **rastgele erişim bellek**
  - **Dinamik Rastgele erişim Bellek (DRAM)**
- İkincil depolama – ana **bellek** uzantısı, büyük **kalıcı** depolama kapasitesi





# Depolama Yapısı (DevamLılık)

- **Sert Disk Sürücü (HDD)** – manyetik kayıt malzemesi ile kaplı sert metal veya cam plakalar
  - Disk yüzeyi mantıksal olarak **izlerden(tracks)** oluşur. İzler **sektörler** şeklinde alt bölümlere ayrılır
  - **Disk denetleyicisi** aygıt ve bilgisayar arasındaki mantıksal etkileşimi belirler
- **Uçucu olmayan bellek (NVM)** cihazlar – sabit disklerden daha hızlı, kalıcı
  - Çeşitli teknolojiler
  - Kapasite ve performans arttıkça daha popüler hale geliyor, fiyat düşüyor





# Depolama Tanımları ve Notasyon İncelemesi

Bilgisayar depolamanın temel birimi, **bit** . Bir bit iki durumdan birini içerebilir değerler, 0 ve 1. Bir bilgisayardaki diğer tüm depolama, bit koleksiyonlarını temel haline dayanır. Yeterince bit verildiğinde, bir bilgisayarın birçok şeyi temsil edebileceği şaşırtıcıdır: sayılar, harfler, resimler, filmler, sesler, belgeler ve programlar, isim sadece bir kaçı. Bir **bayt** 8 bittir ve çoğu bilgisayarda en küçük kullanışlı depolama birimidir. Örneğin, çoğu bilgisayar bir biti taşıyamaz ama bir baytı taşımak için talimatları vardır. Daha az yaygın bir terim, **kelime**, belirli bir bilgisayar mimarisinin yerel veri birimidir. Bir kelime bir veya daha fazla bayt kadardır. Örneğin, 64 bit kaydedicileri olan bir bilgisayar ve 64 bit bellek adresleme genellikle 64 bit (8 bayt) sözcüklere sahiptir. Bilgisayar birçok işlemi aynı anda bayt yerine yerel sözcük boyutunda yürütür.

Bilgisayar depolama, çoğu bilgisayar aktarım hızıyla birlikte geneldir. Bayt ve bayt koleksiyonları işlem ölçü birimidir. A **kilobayt** veya KB , 1.024 bayttır; a **megabayt** veya **MB** ,  $1,024^2$  bayt; a **gigabayt** veya GB ,  $1,024^3$  bayt; a **terabayt** veya **TB** ,  $1,024^4$  bayt; ve bir **petabayt** veya **PB** ,  $1,024^5$  Bayt. Bilgisayar üreticileri genellikle bu sayıları yuvarlar ve megabayt 1 milyon bayt ve gigabayt 1 milyar bayttır. Ağ ölçümler bu genel kuralın bir istisnasıdır; bunlar bit olarak verilir (çünkü ağlar verileri bir anda bir bit olarak taşır).





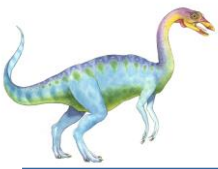


# Depolama Hiyerarşisi

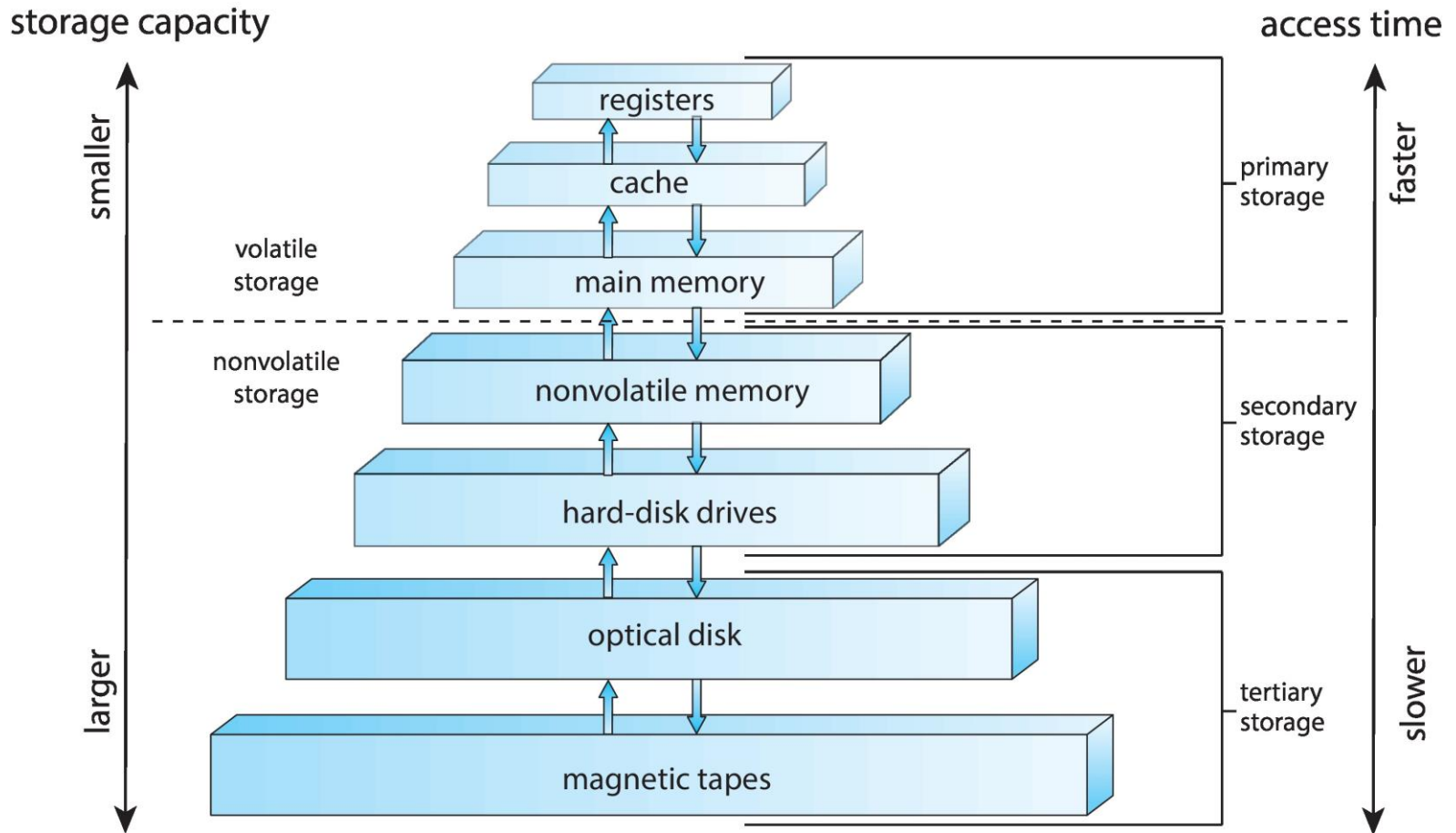
---

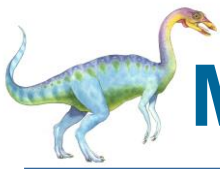
- Hiyerarşik düzenlenmiş depolama sistemleri
  - Hız
  - Maliyet
  - Uçuculuk
- **Önbelleğe alma** – bilgilerin daha hızlı depolama sistemine kopyalanması; ana bellek ikincil depolama için bir önbellek olarak görülebilir
- **Aygıt Sürücüsü** I/O işlemini yönetmek için her aygıt denetleyicisi başına bir sistem yazılımıdır
  - Denetleyici ve çekirdek arasında arabirim sağlar



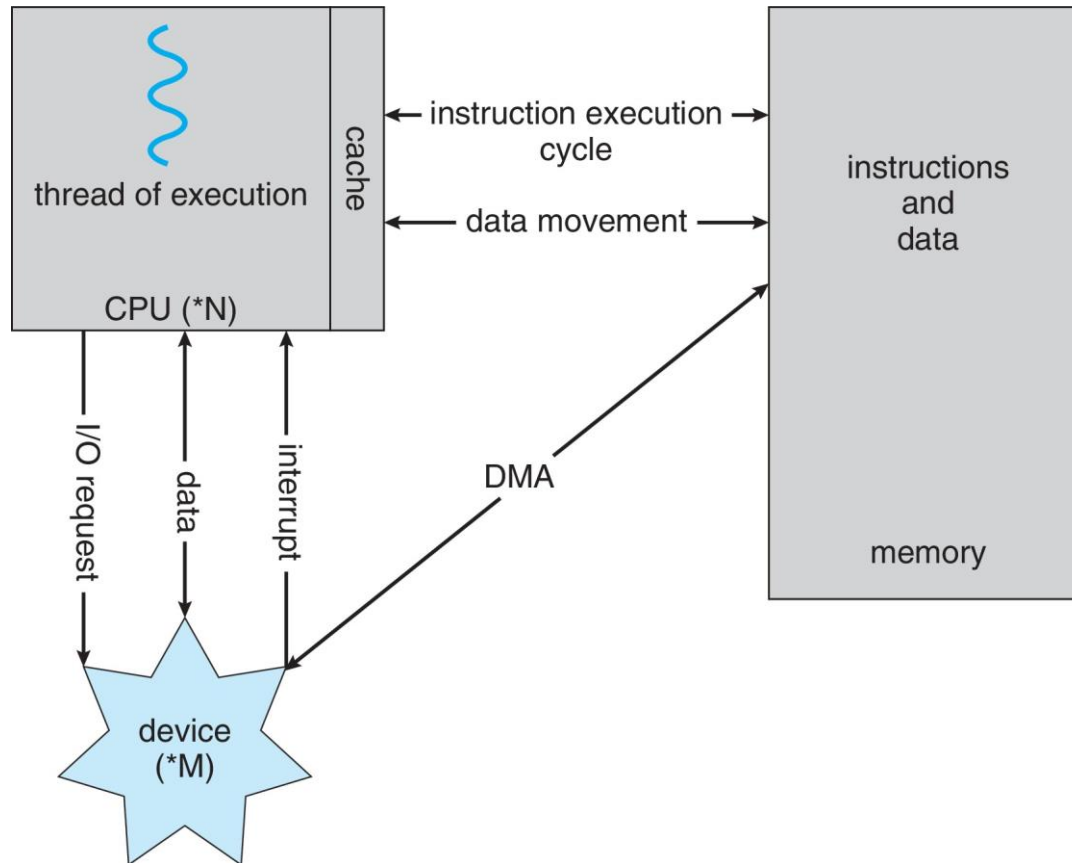


# Depolama Aygıtı Hiyerarşisi



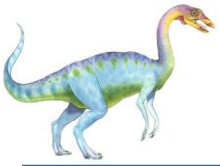


# Modern Bir Bilgisayar Nasıl Çalışır?



*Bir Von Neumann mimarisi*



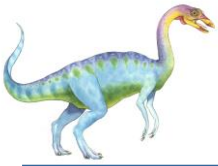


# Doğrudan Bellek Erişim Yapısı

---

- Bellek hızlarına yakın bilgi iletebilen yüksek hızlı I/O cihazları için kullanılır
- Aygıt denetleyicisi, CPU müdahalesi olmadan veri bloklarını arabelleğe depolamadan doğrudan ana belleğe aktarır
- Bayt başına bir kesme yerine blok başına yalnızca bir kesme oluşturulur





# İşletim Sistemi İşlemleri

- Önyükleme(Bootstrap) programı - sistemi başlatmak için basit kod, çekirdeği yükleme
- Çekirdek yüklenir
- **Sistem daemons** (çekirdek dışında sağlanan hizmetler) başlatılır
- Çekirdek **kesme-güdümlüdür** (donanım ve yazılım)
  - Aygıtlardan biri tarafından donanım kesmesi
  - Yazılım kesmesi (**istisna** veya **hata bildirimi**):
    - ▶ Yazılım hatası (örneğin, sıfıra bölme)
    - ▶ İşletim sistemi hizmeti isteği – **sistem çağırısı**
    - ▶ Diğer işlem sorunları arasında sonsuz döngü, birbirini veya işletim sistemini değiştiren prosesler bulunur





# Çoklu programlama (Toplu iş sistemi- Batch)

---

- Tek kullanıcı CPU ve I/O aygıtlarını her zaman meşgul edemez
- Çoklu programlama işleri-jobs (kod ve veri) düzenler, böylece CPU'nun her zaman yürütülecek bir işi vardır
- Sistemdeki toplam işlerin bir alt kümesi bellekte tutulur
- **İş Sıralama(Job Scheduling)** ile bir iş seçilir ve çalıştırılır
- Bir iş beklemek zorunda kaldıysa (örneğin, I/O için), işletim sistemi başka bir işe geçer

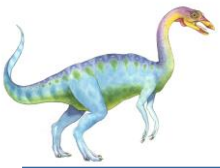




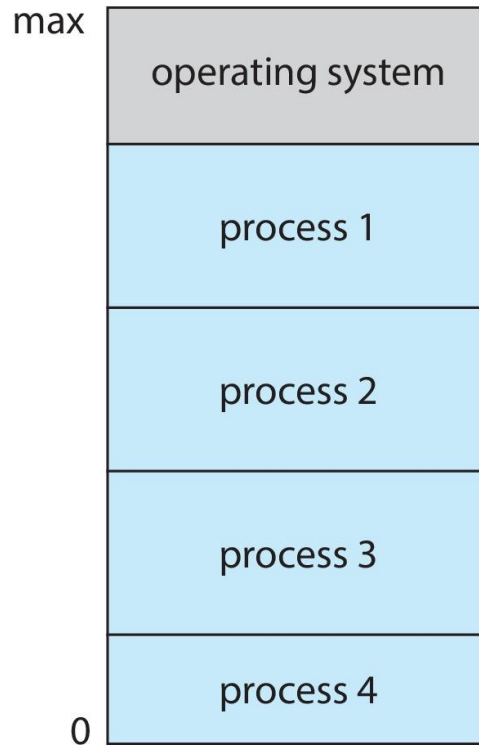
# Çoklu görev (Zaman Paylaşımı)

- Batch sistemlerinin mantıksal bir uzantısı - CPU işleri o kadar sık değiştirir ki, kullanıcılar çalışırken her işle etkileşim kurabilir ve böylece **etkileşimli** bir bilgisayar çalışması mümkün olur
  - **Yanıt süresi** 1 saniye < olmalıdır
  - Her kullanıcının bellekte yürütülen en az bir programı vardır, bu program **proses olarak** adlandırılır
  - Aynı anda çalışmaya hazır birkaç iş  $\Rightarrow$  **CPU Sıralama**
  - İşlemler belleğe sığmaz, **Takas (Swapping) işlemi** prosesleri ramdan içeri veya dışarı taşır
  - **Sanal bellek** tüm parçaları bellekte olmayan işlemlerin yürütülmesine izin verir





# Çok Programlı Sistem için Bellek Düzeni



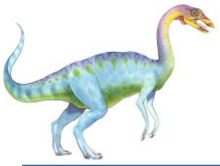




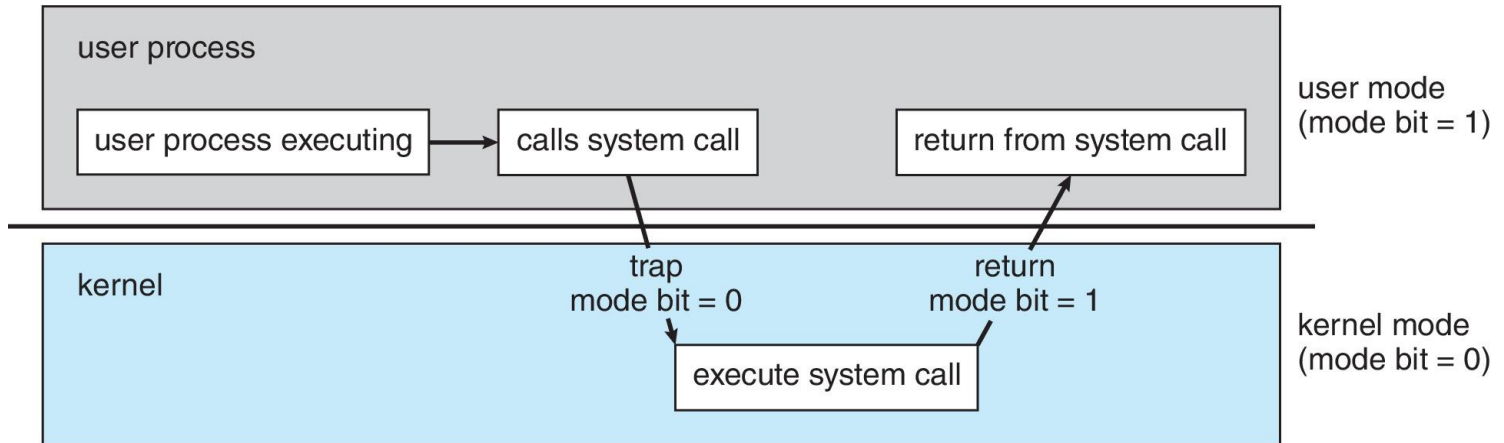
# Çift Modlu Çalışma

- **Çift modlu** çalışma, işletim sisteminin kendisini ve diğer sistem bileşenlerini korumasını sağlar
  - **Kullanıcı modu** ve **çekirdek modu**
- **Mod biti** donanım tarafından sağlanır
  - Sistemin kullanıcı modu veya çekirdek modunda mı olduğunu ayırt etme olanağı sağlar.
  - Kullanıcı çalışırken → mod biti "kullanıcı"
  - Çekirdek kodu yürütülürken → mod biti "çekirdek"
- Bazı talimatlar **ayrıcalıklı**, yalnızca çekirdek modunda yürütülebilir





# Kullanıcıdan Çekirdek Moduna Geçiş

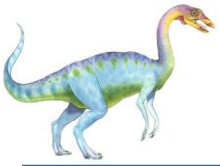




# Proses Yönetimi

- Bir proses, yürütülen bir programdır. Sistem içinde bir çalışma birimidir. Program bir ***pasif varlık***; işlem bir ***aktif varlık***.
- Prosesin görevini yerine getirmek için kaynaklara ihtiyacı var
  - CPU, bellek, I/O, dosyalar
  - Başlatma verileri
- Proses sonlandırma, yeniden kullanılabilir kaynakların geri iadesini gerektirir
- Tek iş parçacıklı proste bir tane **program sayacı** vardır. Bu sayaç yürütülecek bir sonraki yönergenin konumunu tutar.
  - Proses, tamamlanana kadar yönergeleri sırayla, teker teker yürütür
- Çok iş parçacıklı proste iş parçacığı başına bir program sayacı vardır
- Genellikle, sistemde bir veya daha fazla CPU'da aynı anda çalışan birçok proses vardır.
  - Eşzamanlılık CPU'ları prosesler / iş parçacıkları arasında dağıtır





# Proses Yönetimi Faaliyetleri

---

İşletim sistemi, proses yönetimiyle bağlantılı olarak aşağıdaki faaliyetlerden sorumludur:

- Hem kullanıcı hem de sistem prosesleri oluşturma ve silme
- Prosesleri askıya alma ve sürdürme
- Proses senkronizasyonu için mekanizmalar sağlama
- Proses iletişimi için mekanizmalar sağlama
- Kilitlenme yönetimi için mekanizmaları sağlama





# Bellek Yönetimi

- Bir programı yürütmek için yönergelerin/talimatların tümü (veya bir kısmı) bellekte olmalıdır
- Programın ihtiyaç duyduğu verilerin tümü (veya bir kısmı) bellekte olmalıdır
- Bellek yönetimi bellekte neyin ne zaman olması gerektiğini belirler
  - Cpu kullanımı ve kullanıcılara bilgisayar cevap zamanını optimize eder
- Bellek yönetimi faaliyetleri
  - Belleğin hangi bölümlerinin şu anda kullanıldığını ve kim tarafından kullanıldığını izleme
  - Hangi proseslerin (veya bunların parçalarının) ve verilerin belleğe girip çıkacağına karar verme
  - Bellek alanını gerektiği gibi tahsis etme ve iade etme





# Dosya Sistemi Yönetimi

- İşletim sistemi, bilgi depolamanın mantıksal görünümünü sağlar
  - Fiziksel özellikleri mantıksal depolama birimine yansıtır- **dosya**
  - Her ortam aygıt tarafından kontrol edilir (örneğin, disk sürücüsü, teyp sürücüsü)
    - ▶ Değişen özellikler arasında erişim hızı, kapasite, veri aktarım hızı, erişim yöntemi (sıralı veya rastgele) bulunur
- Dosya Sistemi yönetimi
  - Genellikle dizinler halinde düzenlenmiş dosyalar
  - Kimin neye erişebileceğini belirlemek için çoğu sistemde erişim denetimi
  - OS faaliyetleri şunlardır:
    - ▶ Dosya ve dizin oluşturma ve silme
    - ▶ Dosyaları ve dizinleri işlemek için altyapı
    - ▶ Dosyaları ikincil depolama alanına eşleme
    - ▶ Dosyaları kararlı (geçici olmayan) depolama ortamına yedekleme





# Yığın Depolama Yönetimi

---

- Genelde, ana belleğe sığmayan veya "uzun" zaman dilimi tutulması gereken veriler için diskler kullanılır
- Doğru yönetim merkezi öneme sahiptir
- Bilgisayar çalışmasının tüm hızı disk alt sistemine ve algoritmalarına bağlıdır
- OS etkinlikleri
  - Takma ve çıkarma
  - Boş alan yönetimi
  - Depolama ayırma
  - Disk zamanlaması
  - Bölümleme
  - Koruma





# Önbellek (Caching)

- Bir bilgisayarda birçok düzeyde gerçekleştirilen önemli prensip (donanım, işletim sistemi, yazılım)
- Kullanımdaki bilgiler geçici olarak daha yavaştan daha hızlı depolamaya kopyalanır
- Bilgilerin olup olmadığını belirlemek için önce daha hızlı depolama (önbellek) denetlenir
  - Varsa, doğrudan önbellekten kullanılır (hızlı)
  - Yoksa, veriler önbelleğe kopyalanır ve kullanılır
- Önbelleğe alınacak olan bilgiden daha küçük önbellek
  - Önbellek yönetimi önemli tasarım sorunu
  - Önbellek boyutu ve yer değiştirme politikası





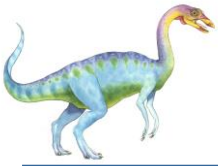


# Çeşitli Depolama Türlerinin Özellikleri

Level	1	2	3	4	5
Name	registers	cache	main memory	solid-state disk	magnetic disk
Typical size	< 1 KB	< 16MB	< 64GB	< 1 TB	< 10 TB
Implementation technology	custom memory with multiple ports CMOS	on-chip or off-chip CMOS SRAM	CMOS SRAM	flash memory	magnetic disk
Access time (ns)	0.25-0.5	0.5-25	80-250	25,000-50,000	5,000,000
Bandwidth (MB/sec)	20,000-100,000	5,000-10,000	1,000-5,000	500	20-150
Managed by	compiler	hardware	operating system	operating system	operating system
Backed by	cache	main memory	disk	disk	disk or tape

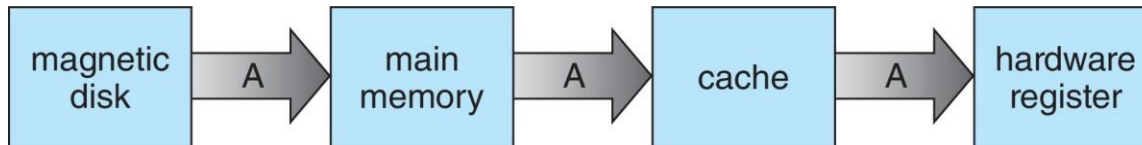
Depolama hiyerarşisi düzeyleri arasındaki hareket açık veya örtük olabilir





# "A" verisinin Diskten Kaydediciye Göçü

- Çoklu görev ortamları, depolama hiyerarşisinde nerede depolanırsa depolansın, en son değeri kullanmaya dikkat etmelidir



- Çok işlemcili ortam donanımında **önbellek tutarlılığı** sağlamalıdır. Öyle ki, tüm CPU'lar önbelleklerinde en güncel değere sahip olmalıdır
- Dağıtılmış ortam durumu daha da karmaşık
  - Bir veriden birkaç kopya olabilir

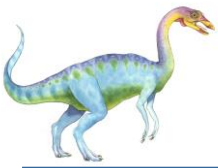




# I/O Alt Sistemi

- İşletim sisteminin bir amacı, donanım cihazlarının özelliklerini kullanıcıdan gizlemektir.
- I/O alt sistemi aşağıdakilerden sorumludur
  - I/O bellek yönetimi
    - ▶ Tamponlama: veriler aktarılırken geçici olarak depolama,
    - ▶ Önbelleğe alma: verilerin parçalarını performans için daha hızlı depolamada tutma,
    - ▶ Biriktirme: bir işin çıktısının diğer işlerin girişiyle çakışması
  - Genel aygıt sürücüsü arabirimi
  - Belirli donanım aygıtlarının sürücüleri



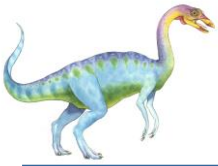


# Koruma ve Güvenlik

---

- **Koruma** – işletim sistemi tarafından tanımlanan kaynaklara proseslerin veya kullanıcıların erişimini kontrol etme mekanizması
- **Güvenlik** – sistemin iç ve dış saldırılara karşı savunulması
  - Hizmet engelleme (denial of service), solucanlar, virüsler, kimlik hırsızlığı, servis hırsızlığı vb.

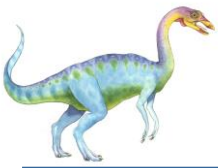




# Koruma

- Sistemler genellikle ilk olarak kimin ne yapacağını belirlemek için kullanıcılar arasında ayrım yapar,
  - Kullanıcı kimlikleri (**kullanıcı ID**, güvenlik IDleri) kullanıcı başına bir ad ve ilişkili numara içerir
  - Kullanıcı ID daha sonra erişim denetimini belirlemek için bu kullanıcının tüm dosyalarıyla, prosesleriyle ilişkilendirilir
  - Grup tanımlayıcısı (**grup ID**) kullanıcı kümesinin tanımlanmasına ve denetimlerin yönetilmesine izin verir, daha sonra her prosesle, dosyayla da ilişkilendirilir
  - **Ayrıcalık yükseltme** kullanıcının haklarını artırır





# Sanallaştırma

- İşletim sistemlerinin diğer işletim sistemlerindeki uygulamaları çalıştırmasına izin verir
  - Geniş ve büyüyen endüstri
- **Emülasyon** Kaynak CPU türü hedef türünden farklı olduğunda kullanılır (örneğin, PowerPC'den Intel x86'ya)
  - Genellikle en yavaş yöntem
  - Bilgisayar dili yerel koda derlenmediği zaman – **Yorumlama**
- **Sanallaştırma** – İşletim sistemi CPU için yerel olarak derlenmiştir, çalışan **misafir** işletim sistemi de yerel olarak derlenmiştir
  - WinXP misafirlerini çalıştıran VMware'i düşünün, her çalışan uygulama, hepsi yerel WinXP **ev sahibi** OS
  - **VMM** (sanal makine Yöneticisi) sanallaştırma hizmetleri sağlar





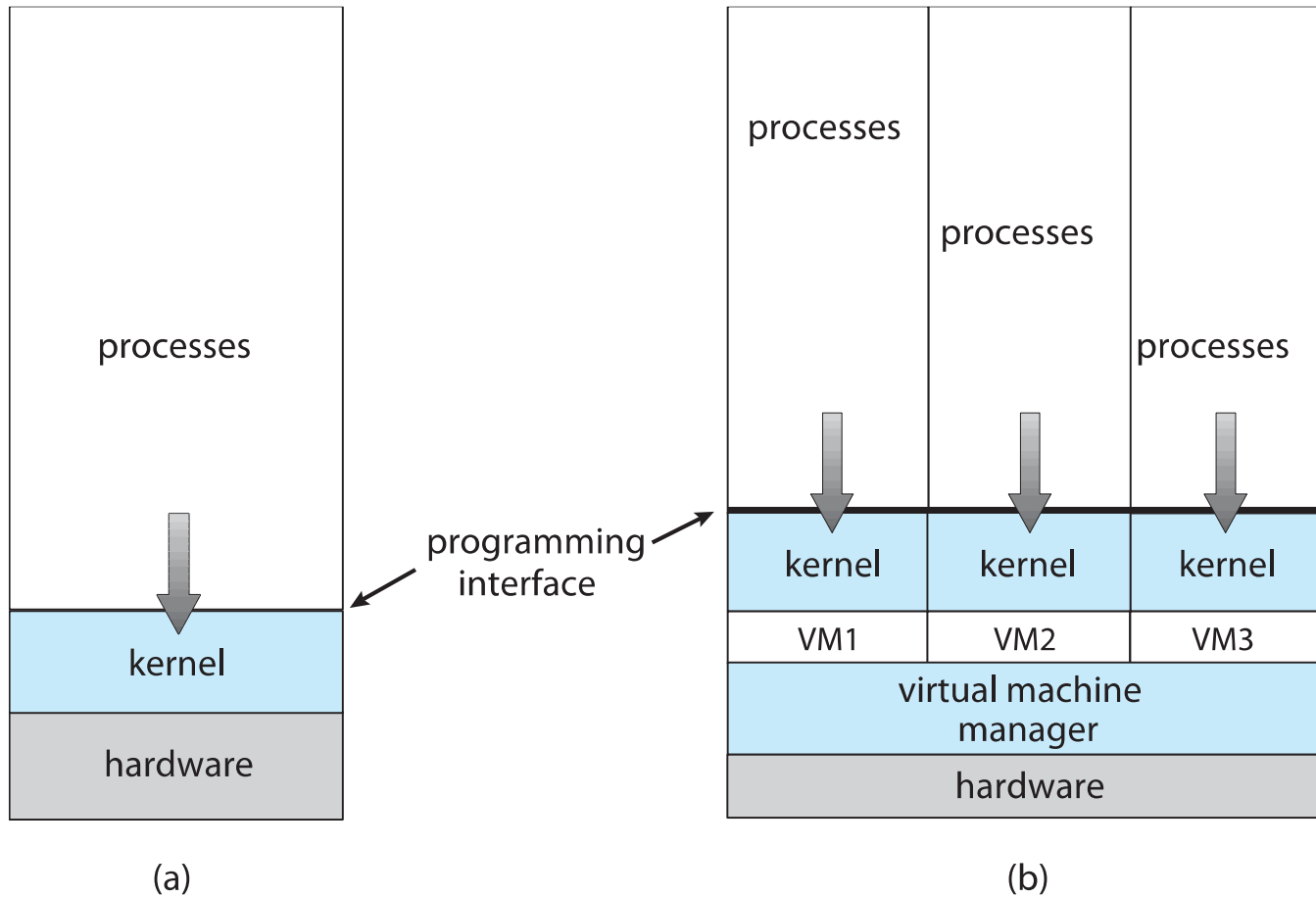
# Sanallaştırma (devam)

- Kullanım örnekleri, keşif veya uyumluluk için birden fazla işletim sistemi çalıştıran dizüstü bilgisayarları ve masaüstlerini kapsar
  - Mac OS X ana bilgisayar çalıştıran Apple dizüstü bilgisayar, konuk olarak Windows
  - Birden fazla sisteme sahip olmadan birden fazla işletim sistemi için uygulama geliştirme
  - Birden fazla sisteme sahip olmadan kalite güvence test uygulamaları
  - Veri merkezlerinde işlem ortamlarını yürütme ve yönetme
- VMM yerel olarak çalışabilir, bu durumda aynı zamanda ana bilgisayardır
  - O zaman genel amaçlı bir ana bilgisayar yoktur (VMware ESX ve Citrix XenServer)

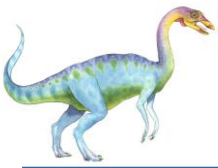




# Sanallaştırma(devam)







# Dağıtılmış Sistemler

- Ayrı, muhtemelen heterojen sistemler koleksiyonunun birbirine ağ şeklinde bağlanması
  - Ağ bir iletişim yoludur, **TCP/IP** en yaygın
    - ▶ Yerel Alan Ağ (LAN)
    - ▶ Geniş Alan Ağ (WAN)
    - ▶ Büyükşehir Alan Ağ (MAN)
    - ▶ Kişisel Alan Ağ (TAVA)
- **Ağ İşletim Sistem** ağ üzerinden sistemler arasında faydalar sağlar
  - İletişim şeması, sistemlerin mesaj alışverişi yapmasına izin verir
  - Tek bir sistem illüzyonu sağlar





# Bilgisayar Sistemi Mimarisi

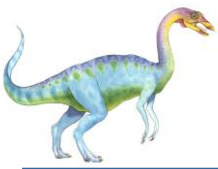




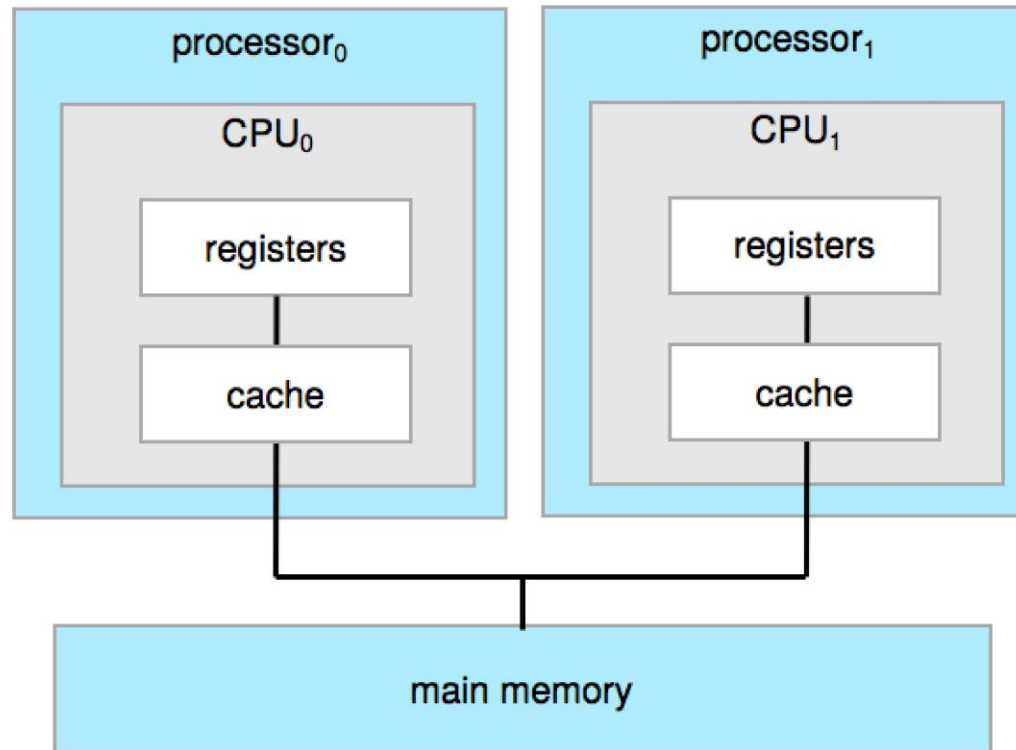
# Bilgisayar Sistemi Mimarisi

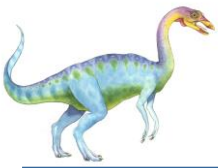
- Çoğu sistem tek bir genel amaçlı işlemci kullanır
  - Çoğu sistemin özel amaçlı işlemcileri de vardır
- **Çok işlemcili** sistemlerin kullanımı ve önemi artmaktadır
  - Ayrıca **paralel sistemler**, **sıkıca bağlı sistemler** olarak adlandırılır
  - Avantajları şunlardır:
    1. **Artan verim**
    2. **Ölçeklememe**
    3. **Artan güvenilirlik** – dereceli bozulma veya hataya dayanıklılık
  - İki tür:
    1. **Asimetrik Çokişlemcili** – her işlemciye özel bir görev atanır.
    2. **Simetrik Çokişlemcili** – her işlemci tüm görevleri yerine getirir





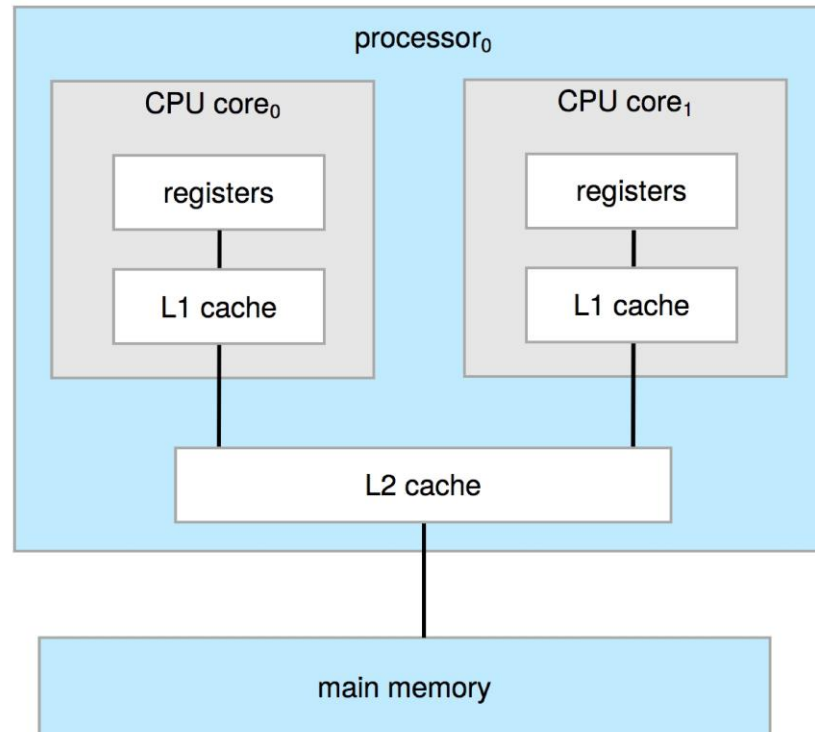
# Simetrik Çok İşlemeli Mimari





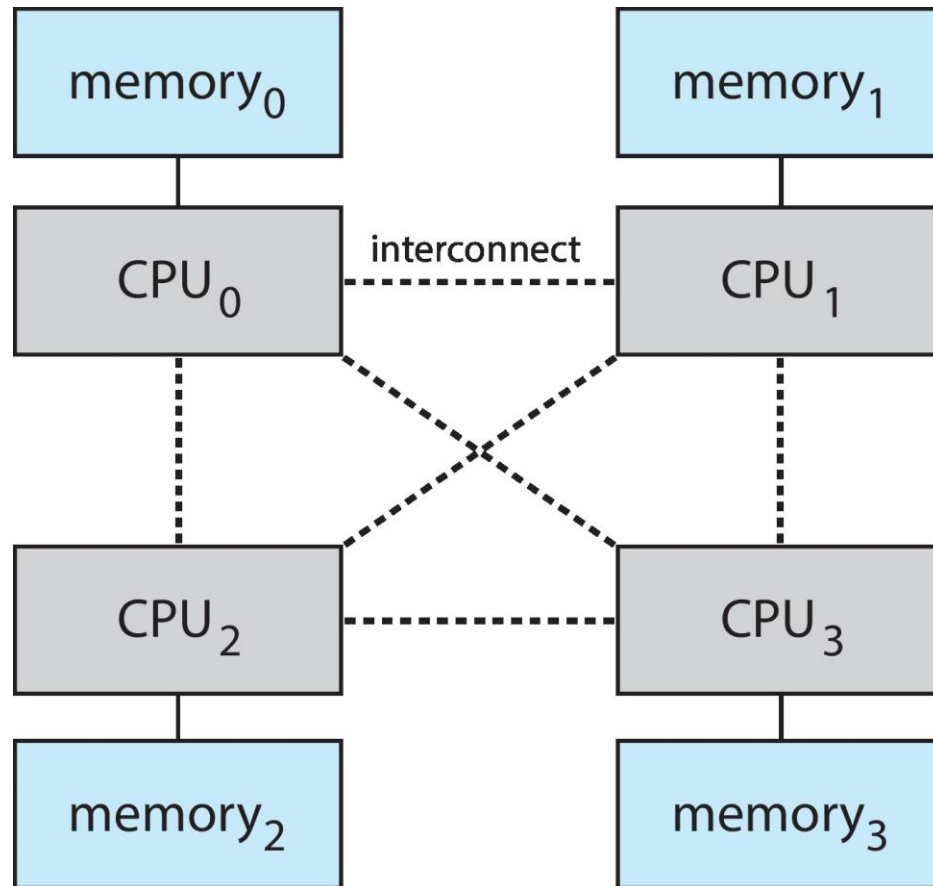
# Çift Çekirdekli Tasarım

- Çok çipli ve **Multicore**
- Tüm çipleri içeren sistemler
  - Birden fazla ayrı sistem içeren kasa





# Çokbiçimli Bellek Erişim Sistemi

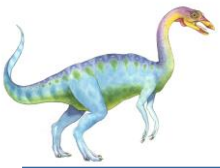




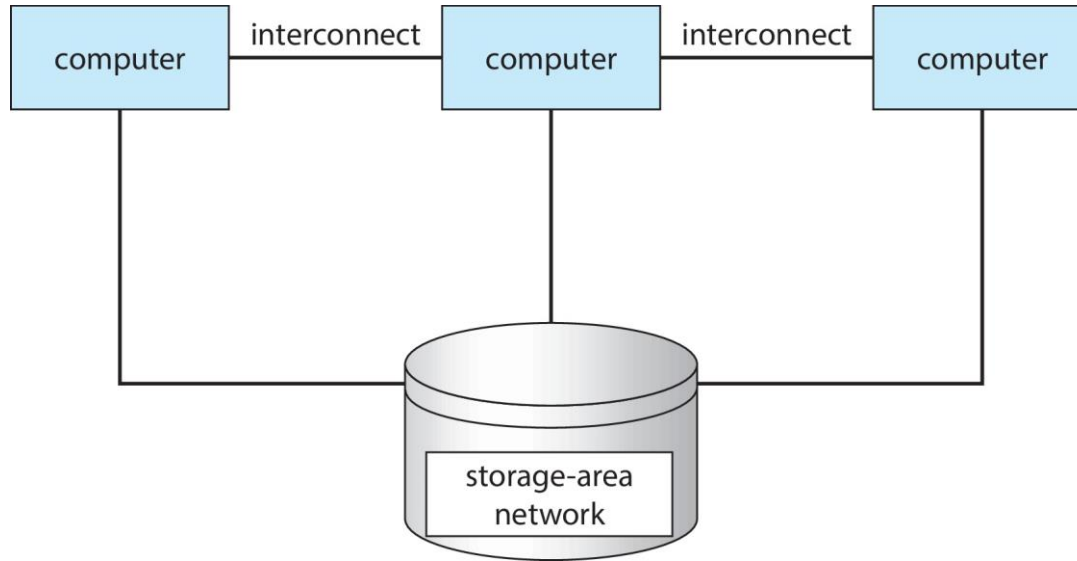
# Kümelenmiş(Clustered) Sistemler

- Çok işlemcili sistemler gibi, ancak birlikte çalışan birden fazla sistem
- Genellikle **depolama alanı ağ (SAN)** ile depolama alanını paylaşırlar
- Hatalar karşısında **yüksek erişime** sahiptir
  - **Asimetrik Kümeleme** bekleme modunda bir makine var
  - **Simetrik Kümeleme** uygulamaları çalıştıran, birbirini izleyen birden çok düğüme sahiptir
- Bazı kümeler : **Yüksek Performans Bilgisayar (HPC)**
  - Uygulamalar **paralel** çalışacak şekilde yazılmalıdır
- Çakışan işlemleri önlemek için **dağıtılmış kilit yöneticisi (DLM)**





# Kümelenmiş Sistemler

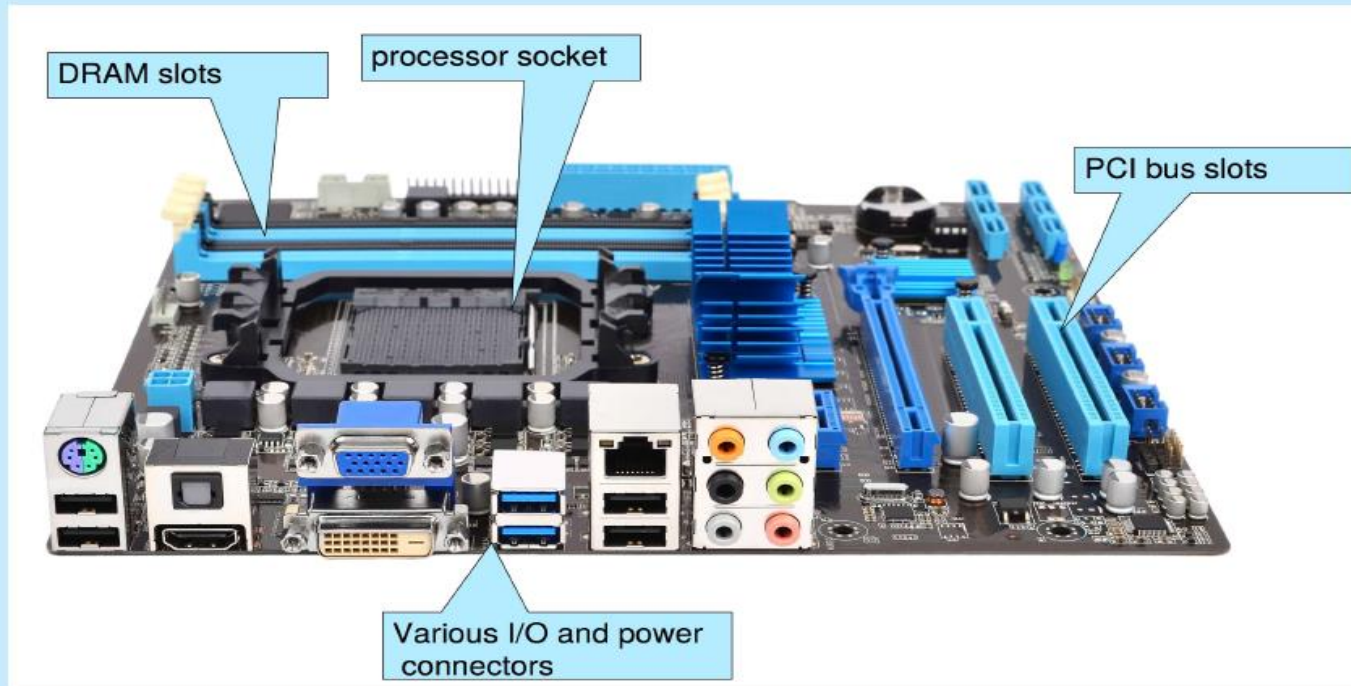






# PC Anakart

Consider the desktop PC motherboard with a processor socket shown below:



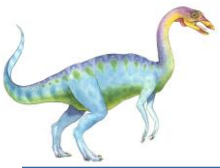
This board is a fully-functioning computer, once its slots are populated. It consists of a processor socket containing a CPU, DRAM sockets, PCIe bus slots, and I/O connectors of various types. Even the lowest-cost general-purpose CPU contains multiple cores. Some motherboards contain multiple processor sockets. More advanced computers allow more than one system board, creating NUMA systems.





# Bilgisayar Sistemi Ortamları



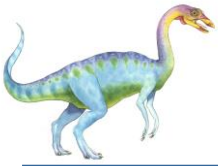


# Bilgi İşlem Ortamları

---

- Geleneksel
- Mobil
- İstemci Sunucu
- Eşler arası (peer to peer)
- Bulut bilişim
- Gerçek Zamanlı Gömülü





# Geleneksel

- Bağımsız genel amaçlı makineler
- Ancak birçok sistem diğerleriyle bağlantı kurarken geleneksel yapıdan uzaklaştı (ör: İnternet)
- **Portal:** sistemlere web erişimi sağlar
- **Ağ Bilgisayarı (ince İstemciler)** Web terminalleri gibidir
- Mobil bilgisayarlar **Kablosuz Ağlar** üzerinden birbirine bağlanır
- Ağ her yerde bulunur hale geliyor - ev sistemleri bile ev bilgisayarlarını İnternet saldırılarına karşı korumak için **güvenlik duvarları** kullanıyor





# Mobil Bilgi İşlem

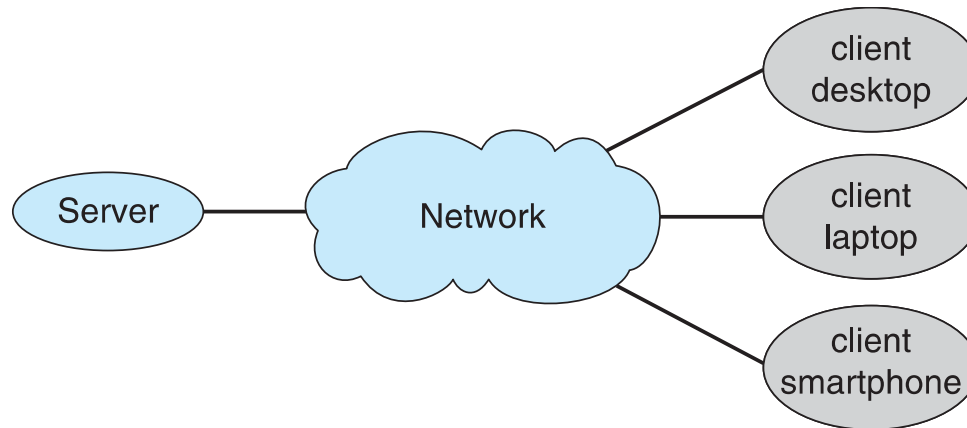
- Akıllı telefonlar, tabletler vb.
- "Geleneksel" bir dizüstü bilgisayarla aralarındaki işlevsel fark nedir?
- Ekstra özellik - daha fazla işletim sistemi özelliği (GPS, jiroskop)
- **artırılmış gerçeklik** gibi yeni uygulama türlerine izin verir
- Bağlantı için IEEE 802.11 kablosuz veya hücresel veri ağlarını kullanır
- Liderler **Apple iOS** ve **Google Android**





# İstemci Sunucu Hesaplaması

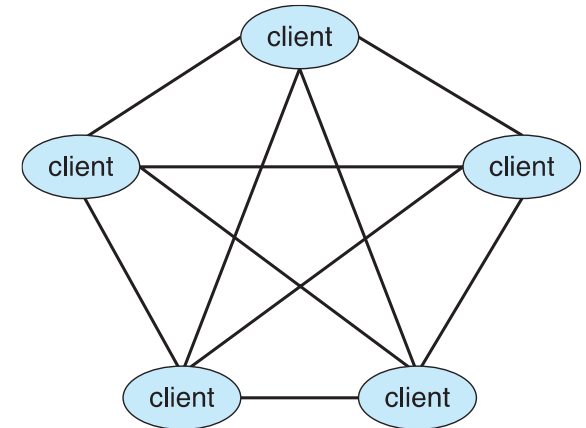
- Akıllı bilgisayarlar tarafından desteklenen aptal terminaller
- Artık birçok sistem **istemci** tarafından oluşturulan isteklere yanıt veren **sunucuya** sahip
  - **İşlem sunucusu sistemi** hizmet talep etmesi için istemciye bir arabirim sağlar (örneğin, veritabanı)
  - **Dosya sunucusu sistemi** İstemcilerin dosyaları depolaması ve alması için arabirim sağlar





# Eşler Arası (peer to peer)

- Başka bir dağıtık sistem modeli
- P2P de istemci / sunucu ayrımı yoktur
  - Bunun yerine, tüm düğümler eş olarak kabul edilir
  - Her biri istemci, sunucu veya her ikisi olarak hareket edebilir
  - Düğüm P2P ağına katılmalı
    - ▶ Hizmetini ağıdaki merkezi arama hizmetiyle kaydeder veya
    - ▶ Hizmet talebini yayınlar ve **keşif protokolü** yoluyla hizmet isteklerine yanıt verir
  - Örnekler Napster ve Gnutella, **Ses üzerinde IP (VoIP)** Skype gibi





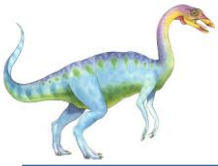
# Bulut Bilişim

---

- Ağ üzerinden hizmet olarak bilgi işlem, depolama ve hatta uygulamalar sunar
- İşlevini yerine getirmek için temel olarak sanallaştırma kullandığından sanallaştırmanın mantıksal bir uzantısıdır
  - Amazon **EC2** binlerce sunucuya, milyonlarca sanal makineye, Internet'te kullanılabilen petabaytlarca depolama alanına sahiptir, kullanıma göre ödeme talep eder



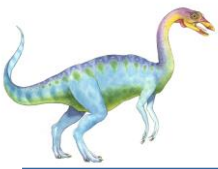




# Bulut Bilişim – Birçok Tipi var

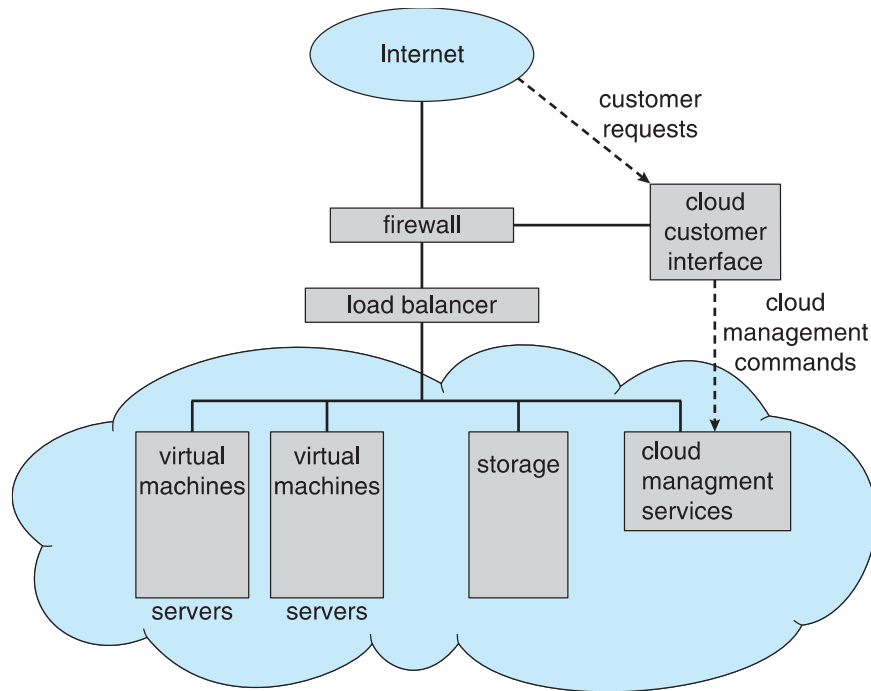
- **Herkese açık bulut** – ödeme yapmak isteyen herkes için İnternet üzerinden kullanılabilir
- **Özel bulut** – şirketin kendi kullanımı için bir şirket tarafından işletilir
- **Melez bulut** – hem genel hem de özel bulut bileşenlerini içerir
- Hizmet Olarak Yazılım (**SaaS**) – İnternet üzerinden kullanılabilen bir veya daha fazla uygulama (örneğin, kelime işlemci)
- Hizmet Olarak Platform (**Paas**) – internet üzerinden uygulama kullanımına hazır yazılım yığını (yani, bir veritabanı sunucusu)
- Hizmet Olarak Altyapı (**IaaS**) – internet üzerinden kullanılabilen sunucular veya depolama (örneğin, yedekleme için kullanılabilir depolama alanı)





# Bulut Bilişim (devam)

- Geleneksel OS ler ve bulut yönetim araçlarından oluşan bulut bilişim ortamları
  - Internet bağlantısı güvenlik duvarları gibi güvenlik gerektirir
  - Yük dengeleyiciler trafiği birden fazla uygulamaya yayar





# Gerçek Zamanlı Gömülü Sistemler

- Gerçek zamanlı gömülü sistemler en yaygın bilgisayar biçimi
  - Önemli, özel amaçlı, sınırlı amaçlı işletim sistemi, **gerçek zamanlı OS**
  - Genişletilebilir
- Diğer birçok özel bilgi işlem ortamı da
  - Bazılarında işletim sistemi vardır, bazılarında işletim sistemi olmadan görev gerçekleştirir
- Gerçek zamanlı işletim sistemi iyi tanımlanmış sabit zaman kısıtlamalarına sahiptir
  - İşleme **olmazsa olmaz** kısıtlama dahilinde yapılmalıdır
  - Yalnızca kısıtlamalar karşılanıyorsa doğru işlem





# Ücretsiz ve Açık Kaynaklı İşletim Sistemleri

- İşletim sistemleri sadece ikili **kapalı kaynak** ve **özel** kılmaktan ziyade, kaynak kodu biçiminde kullanılabilir
- **Kopya koruması** ve **Dijital Haklar Yönetimi (DRM)**
- **Serbest Yazılım Vakıf (FSF)** tarafından başlatıldı, **GNU Kamu Lisans (GPL)**
  - Özgür yazılım ve açık kaynaklı yazılım, farklı insan grupları tarafından savunulan iki farklı fikirdir
    - ▶ <http://gnu.org/philosophy/open-source-misses-the-point.html/>
- Örnekler **GNU/Linux** ve **BSD UNIX** (çekirdek dahil **Mac İŞLETİM SİSTEMİ X**), ve daha fazlası
- VMware Player gibi VMM kullanabilir (Windows'ta Ücretsiz), Virtualbox (açık kaynak kodlu ve birçok platformda ücretsiz - <http://www.virtualbox.com>)
  - Keşif için konuk işletim sistemlerini çalıştırmak için kullanın



# Bölüm 1'in Sonu

