

Bölüm 1: Giriş





Bölüm 1: Giriş

- İşletim sistemleri ne yapar?
- Bilgisayar Sistem Organizasyonu
- Bilgisayar Sistemi Mimarisi
- İşletim Sistemi Yapısı
- İşletim Sistemi İşleyişi
- İşlem Yönetimi
- Bellek (Hafıza) Yönetimi
- Depolama Birimi Yönetimi
- Koruma ve Güvenlik
- Çekirdek Veri Yapıları
- Bilgisayar Ortamları
- Açık Kaynak Kodlu İşletim Sistemleri





Amaç

- Bilgisayar sistemlerinin temel organizasyonunu tanımlamak.
- İşletim sistemlerinin ana bileşenlerini incelemek.
- Birçok bilgisayar ortamına genel bir bakış vermek.
- Açık kaynak kodlu birkaç işletim sistemini keşfetmek.





İşletim Sistemi Nedir?

- Bir bilgisayar kullanıcısı ile bilgisayar donanımı arasında aracı görevi gören bir program.
- İşletim sistemin hedefleri:
 - Kullanıcı programlarını çalıştırmak ve kullanıcı problemlerine kolay çözümler üretmek.
 - Bilgisayar sisteminin kullanımını kolaylaştırmak.
 - Bilgisayar donanımını verimli bir şekilde kullanmak.





İşletim Sistemi

- Anabilgisayar işletim sistemleri, öncelikle donanımın kullanımını optimize etmek için tasarlanmıştır.
- Kişisel bilgisayar (PC) işletim sistemleri, karmaşık oyunları, iş uygulamalar ve insan ve bilgisayar arasındaki diğer programların hepsi.
- Mobil bilgisayarlar için işletim sistemleri, bir kullanıcının programları yürütmek için bilgisayarla kolayca arayüz oluşturabileceği bir ortam sağlar.
- Bu nedenle, bazı işletim sistemleri kullanışlı, bazıları verimli ve bazılarıda ikisinin bir kombinasyonu olacak şekilde tasarlanmıştır.

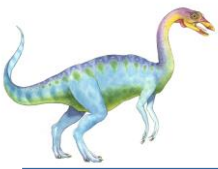




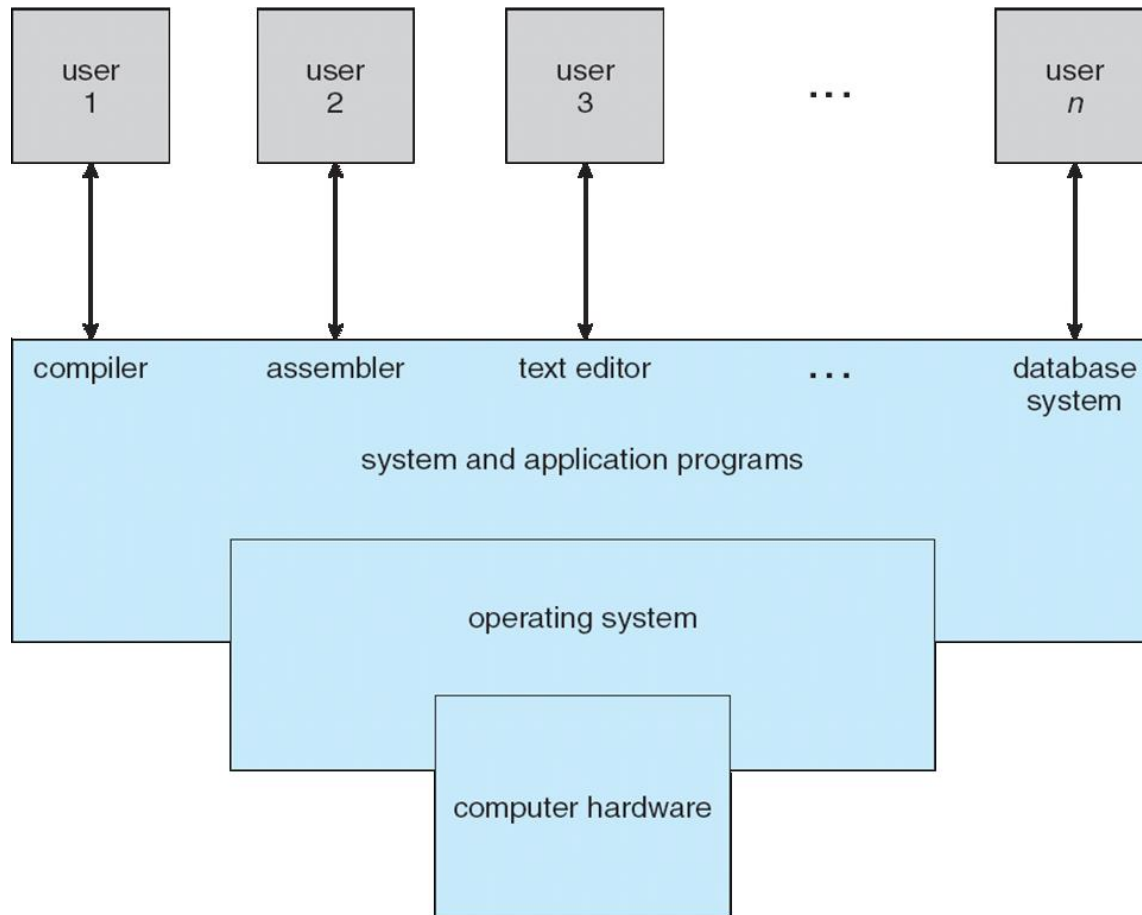
Bilgisayar Sistem Yapısı

- Bilgisayar sistemi dört bileşene ayrılır:
 - **Donanım (hardware)** temel bilişim (computing) kaynaklarını sağlar
 - ▶ İşlemci (CPU), hafıza, I/O aygıtları
 - **İşletim sistemi** çeşitli uygulamalar ve kullanıcılar arasında donanım kullanımını kontrol ve koordine eder.
 - **Uygulama programları** kullanıcıların bilişim problemlerini sistem kaynaklarını kullanarak çözmeye yardımcı olan yazılımlardır.
 - ▶ Kelime işlemciler, derleyiciler, web tarayıcıları, veri tabanı sistemleri, oyunlar
 - **Kullanıcılar**
 - ▶ İnsanlar, makineler, diğer bilgisayarlar





Bilgisayar Sisteminin Dört Bileşeni





İşletim Sistemleri Ne yapar?

- Bakış açısına göre değişir.
- Kullanıcılar **kullanım kolaylığı** ve **iyi performans** isterler.
 - **Kaynak kullanımını** önemsemezler.
- Ancak **ana bilgisayar** veya **mini bilgisayar** gibi paylaşılan bilgisayarlar tüm kullanıcıları mutlu etmelidir.
- **İş istasyonları (workstations)** gibi bağımlı sistemlerin kullanıcıları özel kaynaklara sahiptir, ancak sıklıkla sunuculardan paylaşılan kaynakları kullanırlar.
- El bilgisayarları kaynak yetersizliği, kullanılabilirlik ve pil ömrü için optimize edilmiştir.
- Bazı bilgisayarlarda çok az kullanıcı arabirimi vardır veya yoktur, mesela gömülü bilgisayarlar (embedded computers), otomobiller.





İşletim Sistemleri Ne yapar?

- Bilgisayar sistemlerinin temel amacı, kullanıcı programlarını yürütmek ve kullanıcı problemlerini çözmeyi kolaylaştırmaktır.
- Bilgisayar donanımı bu amaca yönelik olarak inşa edilmiştir.
- Sadece donanımın tek başına kullanımı özellikle kolay olmadığı için uygulama programları geliştirilmektedir.
- Bu programlar, G/Ç cihazlarını kontrol edenler gibi belirli ortak işlemleri gerektirir.
-





İşletim Sistemi Tanımı

- İşletim sistemi **kaynak dağıtıcıdır (resource allocator)**.
 - Tüm kaynakları yönetir.
 - Verimli ve adil kaynak kullanımı için birbiriyle çelişen talepler arasında karar verir.
- İşletim sistemi **kontrol programıdır (control program)**.
 - Programların çalışma hatalarını ve uygunsuz kullanımlarını önlemek için programların yürütülmesini denetler.





İşletim Sistemi Tanımı

- Ancak günümüzde mobil cihazlar için işletim sistemlerine baktığımızda, işletim sistemini oluşturan özelliklerin bir kez daha arttığını görüyoruz.
- Mobil işletim sistemleri genellikle yalnızca bir core kernel değil, aynı zamanda ara katman yazılımları da içerir; bu ara katman yazılımları uygulama geliştiriciler için ek servisler sağlar.





İşletim Sistemi Tanımı (Devam)

- Evrensel kabul gören bir tanımlı yoktur.
- İşletim sistemi üreticisinin bir işletim sistemine dahil ettiği her şeydir. Hemen hemen doğru bir cevap.
 - Fakat teknoloji ile birlikte sürekli değişmektedir.
- Bir bilgisayarda her zaman çalışan tek program **kernel**'dir.
- Diğer her şey:
 - Ya sistem programı (system programs) (işletim sistemi ile birlikte gelir)
 - Yada uygulama programıdır (application programs).





Bilgisayarın Başlatılması

- Bilgisayar yeniden başlatıldığında ya da açıldığında **önyükleyici program (bootstrap program)** çalıştırılır.
 - Tipik olarak ROM veya EPROM'da tutulur ve genellikle **aygıt yazılımı (firmware)** olarak adlandırılır.
 - Sistemi tüm yönleri ile başlatır. **Önyükleyici program**, işletim sistemini nasıl yükleyeceğini ve bu sistemi çalıştırmaya nasıl başlayacağını bilmelidir.
 - İşletim sisteminin çekirdeğini (**kernel**) yükler ve çalıştırır.
 - Kernel sistem programlarını yükler ve çalıştırır.

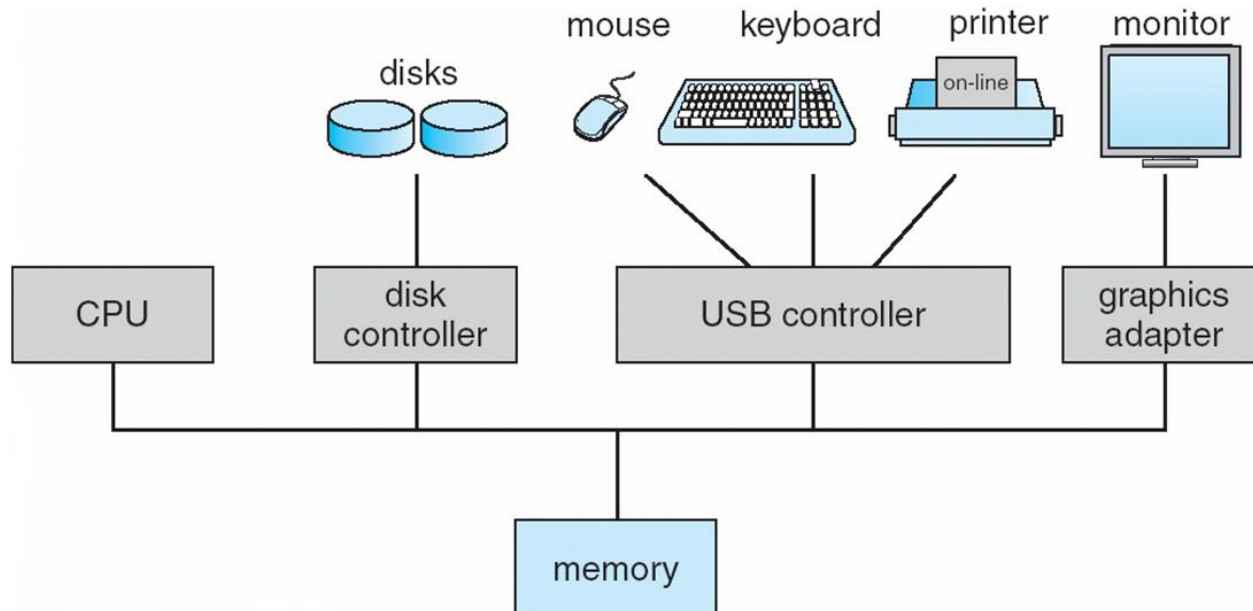




Bilgisayar Sistemi Organizasyonu

■ Bilgisayar sistemin işleyişi

- Bir veya birden fazla CPU ve aygıt denetleyicileri (device controllers) ortak bir veri yolu üzerinden paylaşılan ana hafızaya bağlanır.
- Aynı anda çalışan işlemciler ve cihazlar hafızaya erişmek için birbirleriyle yarışır.





Bilgisayar Sistemi İşleyişi

- I/O aygıtları ve CPU eş zamanlı olarak çalışabilir.
- Her bir cihaz denetleyicisi belli bir tip cihazın kontrolünden sorumludur.
- Her cihaz denetleyicisinin bir **yerel tampon belleği (local buffer)** vardır.
- CPU, ana hafıza ile yerel tampon bellekler arasında çift yönlü veri taşır.
- I/O işlemi, cihazdan, denetleyicinin yerel tampon belleğine doğru gerçekleşir.
- Aygıt denetleyicisi, bir kesme (interrupt) işlemine neden olarak işlemini tamamladığını CPU'ya bildirir.





Kesmelerin Ortak Fonksiyonları

- Kesmelerin kontrolü, (interrupt vector) kesme vektörü aracılığıyla kesme servis rutinine (interrupt service routine) gönderilir.
- Kesme vektöründe hangi servis rutininin hangi hafıza adresinde olduğu bilgisi bulunmaktadır.
- Kesme mimarisi, kesme talimatın adresini kaydetmelidir.
- Bir tuzak (**trap**) veya istisna (**exception**) yazılım temelli bir kesmedir, bir hata veya kullanıcı isteği nedeniyle oluşan yazılım tarafından oluşturulan bir kesintidir.
- İşletim sistemi kesme eğilimlidir (**interrupt driven**).

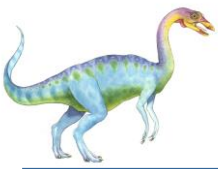




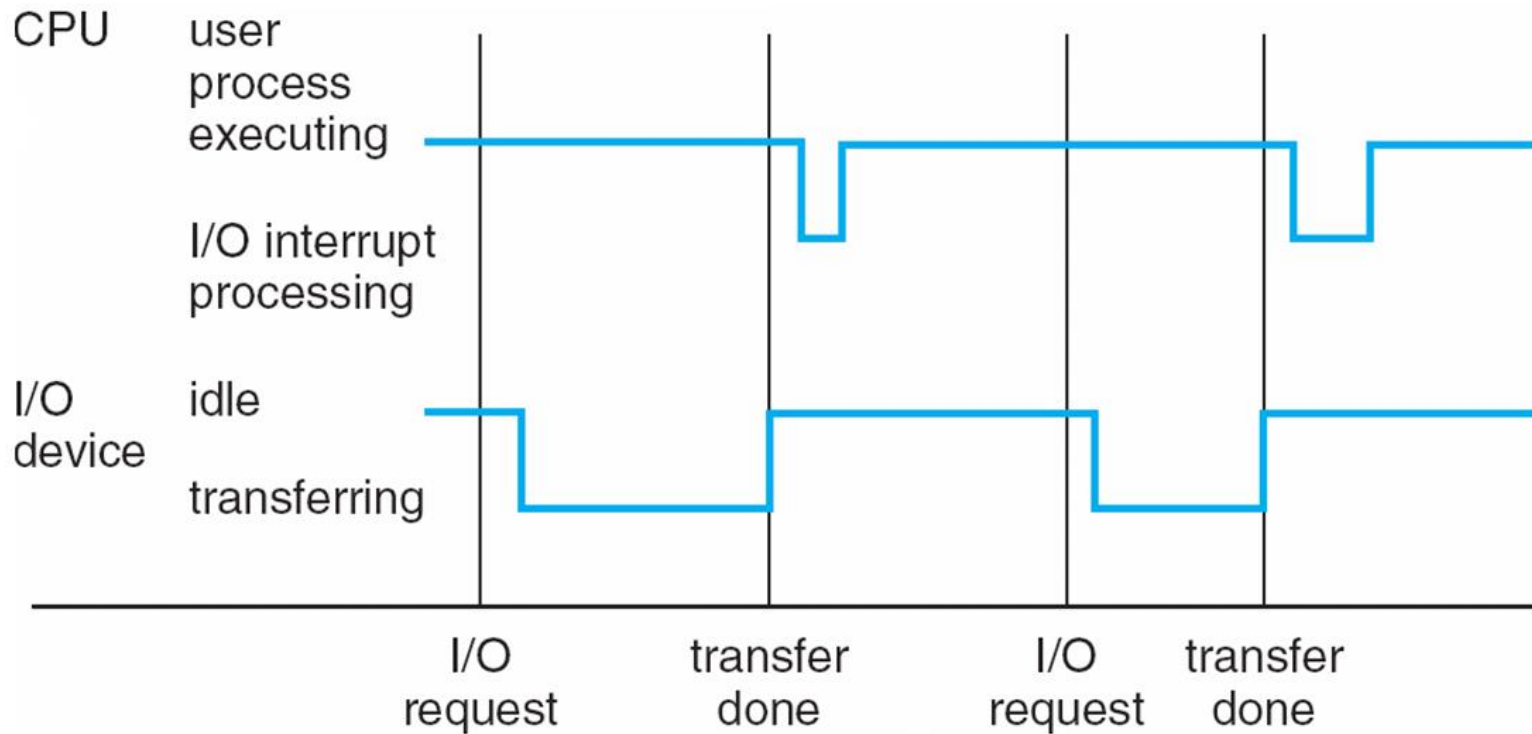
Kesme İşlemi

- İşletim sistemi yazmaçları (register) ve program sayaçlarını hafızaya saklayarak CPU'nun durumunu muhafaza eder.
- Hangi tür kesmenin oluştuğunu belirler:
 - **polling**
 - **vectored** kesme sistemi
- Kernelde bulunan ayrı kod bölümleri, her bir kesme türü için hangi eylemin yapılması gerektiğini belirler.





Kesme Zaman Çizelgesi





Depolama Yapısı

- Ana bellek - CPU'nun doğrudan erişebileceği büyük depolama ortamı
 - Rasgele erişimli (Random access)
 - Geçici (volatile)
- İkincil depolama - büyük kalıcı (**nonvolatile**) depolama kapasitesi sağlayan ana belleğin genişletilmiş halidir.
- Sabit diskler - manyetik kayıt malzemesi ile kaplı sert metal veya cam plakalar
 - Disk yüzeyi mantıksal olarak **track'lere** bölünmüştür. **Tracklerde sectorlere** bölünür.
 - Disk denetleyicisi (**disk controller**), aygıt ve bilgisayar arasındaki mantıksal etkileşimi belirler.
- **Solid-state diskler** harddisklerden daha hızlıdır.
 - Çeşitli teknolojileri vardır.
 - Gün geçtikçe daha popüler hale gelmektedirler.





Depolama Yapısı

- Her depolama sistemi, bir veriyi depolama ve bu veriyi daha sonra alınana kadar tutma temel işlevlerini sağlar.
- Tüm bellek biçimleri bir dizi byte ile tutulur.
- Her byte'ın kendi adresi vardır.
- Bilgilerin yazılıp okunması, belirli bellek adreslerine bir dizi yükleme talimatı (load instruction) veya depolama talimatı (store instruction) yoluyla gerçekleştirilir.
- Yükleme talimatı, bir baytı veya kelimeyi ana bellekten CPU içindeki dahili bir kayıt defterine taşıırken, mağaza talimatı bir kaydın içeriğini ana belleğe taşır.
- Von Neumann mimarisine sahip bir sistemde yürütülen tipik bir talimat-yürütme (instruction-execution) döngüsü, önce bellekten bir talimat alınır ve registerda depolanır. (Fetch)
- Daha sonra talimat çözülür. (Decode) Talimatın çözülmesi ile birlikte operandlar bellekten alınıp registerlara yazılır.
- Operandlar üzerindeki komutlar işlendikten sonra sonuç bellekte saklanabilir.





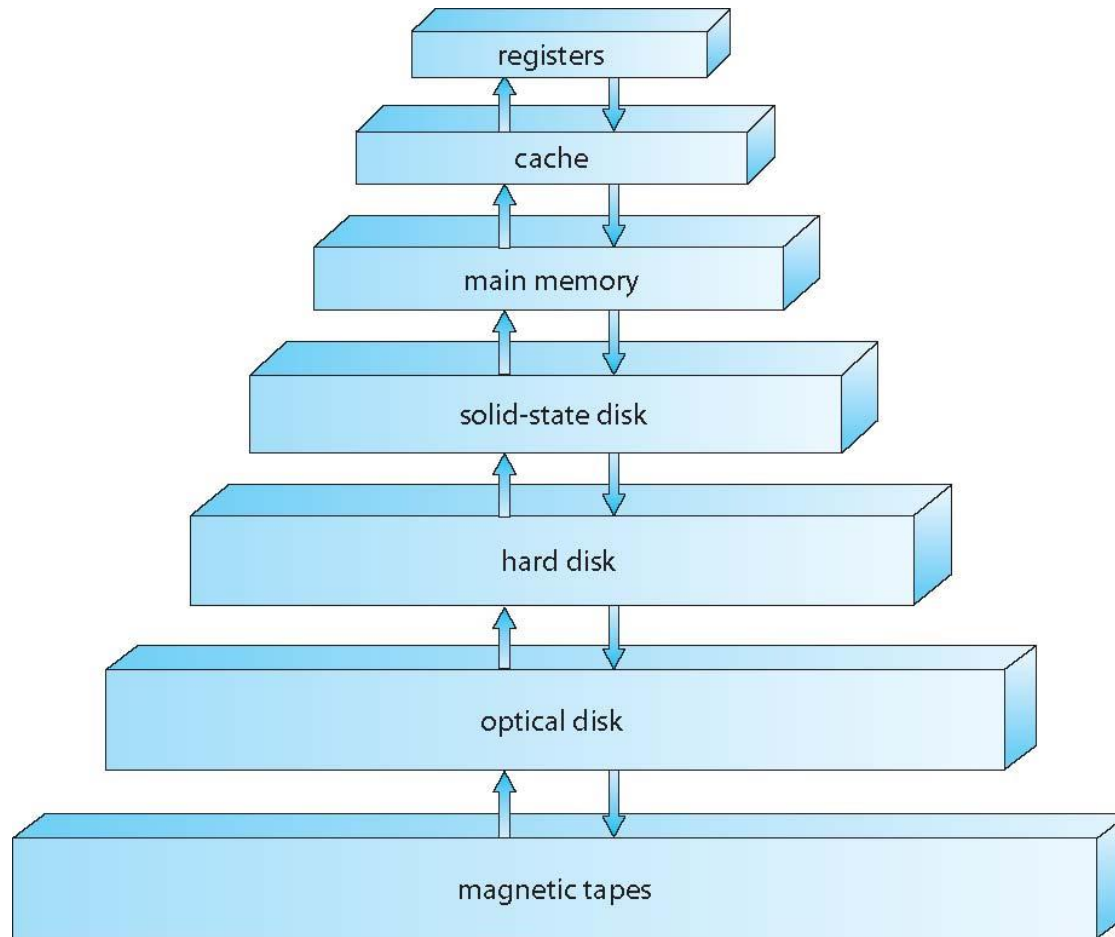
Depolama Hiyerarşisi

- Depolama sistemlerin hiyerarşisinde
 - Hız
 - Maliyet
 - Kalıcılık
- **Önbellek (caching)** bilgileri daha hızlı depolama sistemine kopyalamayı sağlar. Ana bellek ikincil bellek için bir önbellek olarak düşünülebilir.
- Aygıt denetleyicisi I/O işlemini kontrol etmek için **Aygıt sürücüsünü (Device Driver)** kullanır
 - Denetleyici ve çekirdek arasında iletişimi sağlar.





Depolama-Aygıt Hiyerarşisi





Önbellek (Caching)

- Bir bilgisayarda birçok seviyede gerçekleştirilen önemli işlem (donanım, işletim sistemi, yazılım)
- Hızlı depolama aygıtında (cache) bilginin olup olmadığı kontrol edilir.
 - Varsa, bilgi cache'den direk kopyalanır (hızlı)
 - Yoksa, bilgi cache'e kopyalanır ve oradan kullanılır.
- Cache depolama alanından daha küçüktür.
 - Cache yönetimi önemli bir tasarım problemidir
 - Cache büyüklüğü ve yer değiştirme politikası





I/O Yapısı

- Genel amaçlı bir bilgisayar sistemi, ortak bir veri yolu aracılığıyla bağlanan CPU'lardan ve birden çok cihaz denetleyicisinden oluşur.
- Her aygıt denetleyicisi, belirli bir aygıt türünden sorumludur. Denetleyiciye bağlı olarak, birden fazla aygıt eklenebilir. Örneğin, küçük bilgisayar sistemleri arabirimi (Small Computer System Interface) denetleyicisine yedi veya daha fazla aygıt bağlanabilir.
- Bir aygıt denetleyicisi, bazı yerel arabellek depolama ve bir dizi özel amaçlı registerlar barındırır.
- Aygıt denetleyicisi, verileri denetlediği çevresel aygıtlar ile yerel arabellek deposu arasında taşımaktan sorumludur.
- Tipik olarak, işletim sistemlerinde her aygıt denetleyicisi için bir aygıt sürücüsü bulunur. Bu aygıt sürücüsü, aygıt denetleyicisini anlar ve işletim sisteminin geri kalanına aygıt için tek tip bir arabirim sağlar.





I/O Yapısı

- Bir G/Ç işlemini başlatmak için aygıt sürücüsü, aygıt denetleyicisine uygun kayıtları yükler.
- Aygıt denetleyicisi, sırayla, hangi eylemin gerçekleştirileceğini belirlemek için bu kayıtların içeriğini inceler ("klavyeden bir karakter okuma" gibi). Kontrolör, cihazdan yerel arabelleğine veri aktarımını başlatır.
- Veri aktarımı tamamlandıktan sonra, aygıt denetleyicisi, çalışmasının tamamlandığını bir kesme yoluyla aygıt sürücüsüne bildirir. Aygıt sürücüsü daha sonra kontrolü işletim sistemine geri verir, muhtemelen veriyi veya işlem bir okuma ise verilere bir işaretçiye döndürür.
- Diğer işlemler için aygıt sürücüsü durum bilgilerini döndürür.





I/O Yapısı

- Uygulama programları işletim sistemi yardımıyla I/O (bir aygıttan okuma veya bir aygıta yazma) isteğinde bulunabilirler.
 - İstek **Sistem Çağrısı (System Call)** yardımıyla yapılır.
 - İşletim sistemindeki sistem çağrısı rutini, işletim sistemindeki aygıt sürücüsü rutinleri aracılığıyla I/O işlemini gerçekleştirir.
 - İşletim sistemi cihaz durum tablosunu yönetir: Her cihaz başına bir giriş bulunur. Bu girişte cihazın durum bilgisi bulunur.
 - Bir sistem çağrısı verildikten sonra, uygulama aramanın bitmesini bekleyebilir (blocking call/çağrı engelleme) veya başka bir şey yapmaya devam edebilir (non-blocking call).





Direk Hafıza Erişimi Yapısı

- Bu kesme güdümlü G/Ç biçimi, küçük miktarlardaki verilerin taşınması için uygundur, ancak disk G/Ç gibi toplu veri hareketi için kullanıldığında yüksek ek yük oluşturabilir.
- Bu sorunu çözmek için doğrudan bellek erişimi (DMA) kullanılır. G/Ç aygıtı için arabellekleri, işaretçileri ve sayaçları ayarladıktan sonra, aygıt denetleyicisi, CPU'nun müdahalesi olmadan tüm bir veri bloğunu doğrudan kendi arabellek deposundan belleğe aktarır.
- Düşük hızlı aygıtlar için oluşturulan bayt başına bir kesme yerine, aygıt sürücüsüne işlemin tamamlandığını söylemek için blok başına yalnızca bir kesme oluşturulur.
- Aygıt denetleyicisi bu işlemleri gerçekleştirirken, CPU diğer işleri gerçekleştirmek için kullanılabilir.





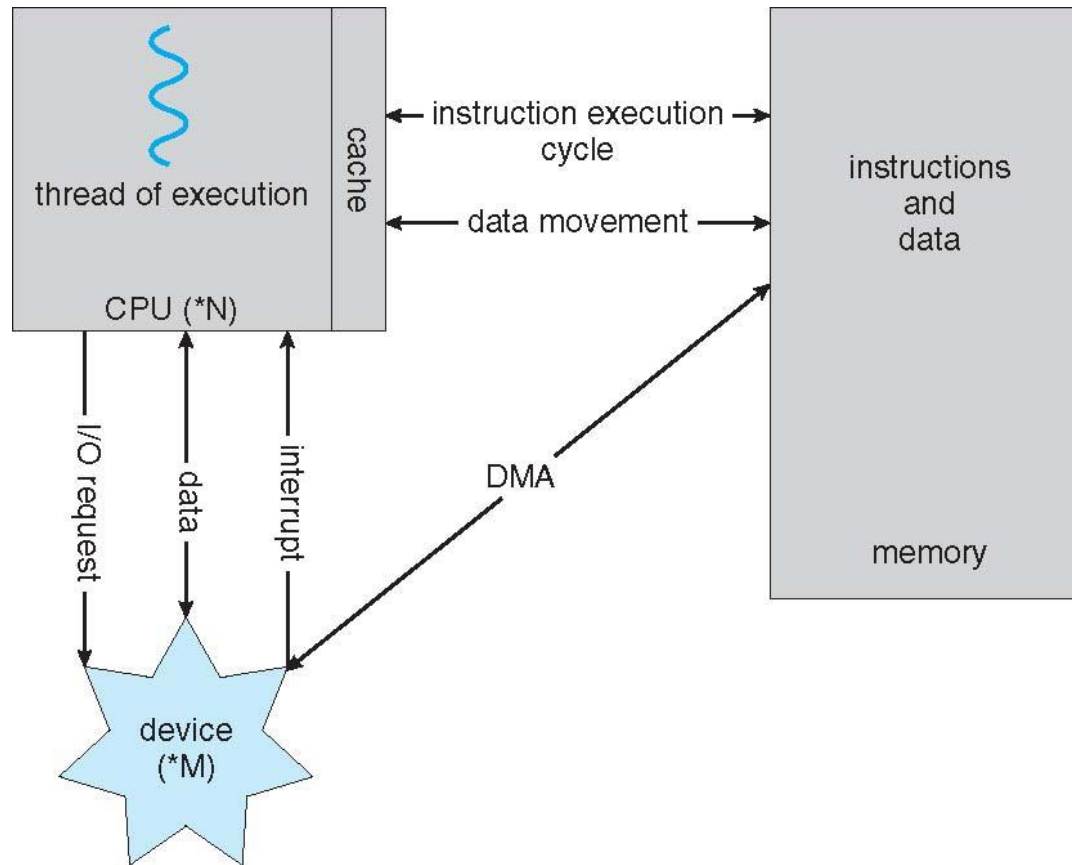
Direk Hafıza Erişimi Yapısı

- Yüksek hıza sahip I/O aygıtları için kullanılır ve bilgiyi hafıza hızına yakında şekilde transfer eder.
- Aygıt denetleyicisi, arabellek depolamasından (buffer) veri bloklarını CPU müdahalesi olmadan doğrudan ana belleğe aktarır.
- Her bayt başına bir kesme yerine, blok başına yalnızca bir kesme oluşturulur.





Modern Bilgisayarlar Nasıl Çalışır?



A von Neumann architecture





Çok İşlemcili Sistemler

- Geçtiğimiz birkaç yıl içinde, çok işlemcili sistemler (paralel sistemler veya çok çekirdekli sistemler olarak da bilinir) bilgi işlem ortamına hakim olmaya başladı.
- Bu tür sistemler, bilgisayar veri yolunu ve bazen saati, belleği ve çevre aygıtlarını paylaşan yakın iletişimde iki veya daha fazla işlemciye sahiptir.
- Çok işlemcili sistemler ilk olarak sunucularda belirgin bir şekilde kullanılmaya başlandı ve ondan sonra heri masaüstü ve dizüstü sistemlere geçti.
- Son zamanlarda, akıllı telefonlar ve tablet bilgisayarlar gibi mobil cihazlarda birden fazla işlemci kullanılmaktadır.





Çok işlemcili Sistemler (Multiprocessors)

- Çok işlemcili sistemlerin üç avantajı:
 - **Artan iş hacmi (verim):** İşlemci sayısını artırarak, daha kısa sürede daha fazla iş yapmak hedefidir. Ancak N işlemci için hızlanma oranı N değildir; N 'den küçüktür. Birden çok işlemci bir görev üzerinde işbirliği yaptığında, tüm parçaların doğru şekilde çalışmasını sağlamak için belirli bir miktar ek yük ortaya çıkar. Bu ek yük ve paylaşılan kaynaklar için çekişme, ek işlemcilerden beklenen kazancı düşürür.
 - **Ekonomi Ölçeği:** Çok işlemcili sistemler, çevre birimlerini, yığın depolamayı ve güç kaynaklarını paylaşabildiklerinden, eşdeğer birden çok tek işlemcili sistemlerden daha ucuza mal olabilir. Birkaç program aynı veri kümesi üzerinde çalışıyorsa, bu verileri tek bir diskte depolamak ve tüm işlemcilerin bunları paylaşmasını sağlamak, yerel disklere ve verilerin birçok kopyasına sahip çok sayıda bilgisayara sahip olmaktan daha ucuzdur.
 - **Artan Güvenlik:** Artan güvenilirlik. İşlevler birkaç işlemci arasında düzgün bir şekilde dağıtılabiliyorsa, bir işlemcinin arızalanması sistemi durdurmaz, yalnızca yavaşlatır. On işlemcimiz varsa ve biri arızalanırsa, kalan dokuz işlemcinin her biri arızalı işlemcinin çalışmasından bir pay alabilir. Böylece tüm sistem yalnızca yüzde 10 daha yavaş çalışır, tamamen başarısız olmaz.





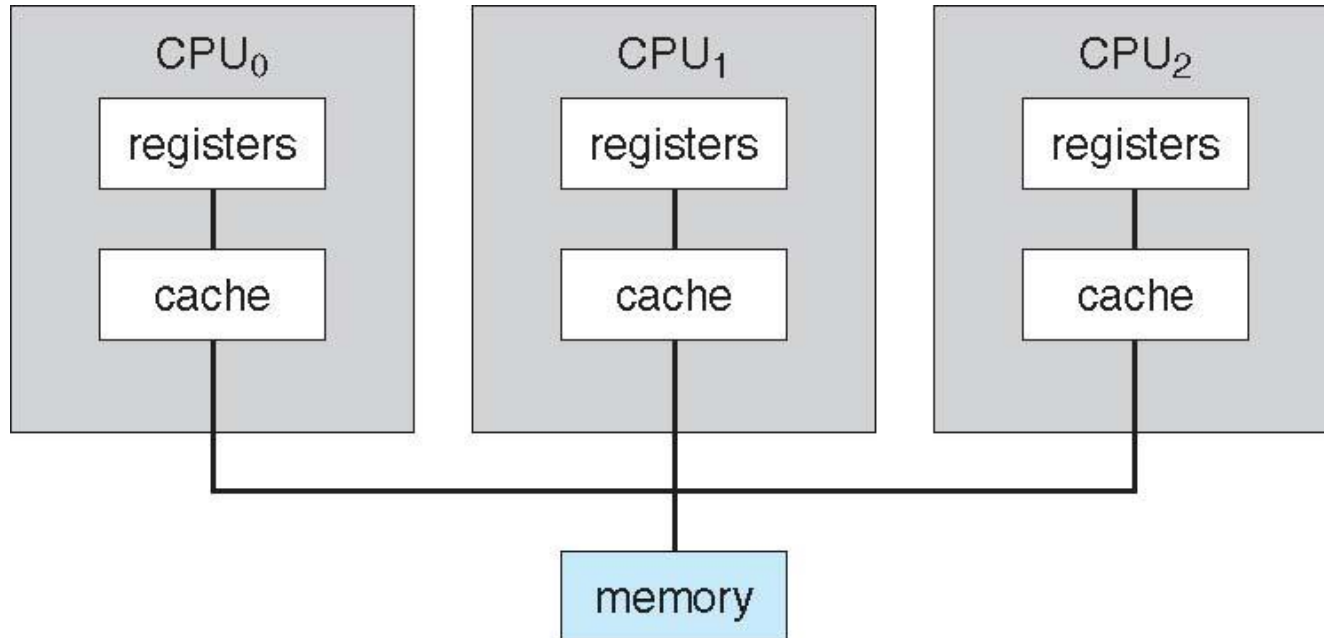
Çok işlemcili Sistemler (Multiprocessors)

- İki türü vardır:
- **Asimetrik çok işlemci (Asymmetric Multiprocessing)** – her işlemciye bir görev atanır. Master – slave. Bir patron işlemci sistemi kontrol eder; diğer işlemciler ya talimat için patrona bakar ya da önceden tanımlanmış görevlere sahiptir. Bu şema bir patron-işçi ilişkisini tanımlar. Patron işlemci, işi çalışan işlemcilere planlar ve tahsis eder.
- **Simetrik çok işlemci (Symmetric Multiprocessing, SMP)** – her işlemci bütün görevleri işletim sistemi ile yerine getirir. SMP, tüm işlemcilerin eş olduğu anlamına gelir; işlemciler arasında patron-işçi ilişkisi yoktur. Her işlemcinin kendi kayıt setinin yanı sıra özel veya yerel bir önbelleğe sahiptir. Ancak, tüm işlemciler fiziksel belleği paylaşır. Bu modelin yararı, birçok işlemin aynı anda çalışabilmesidir - N CPU varsa N işlem çalışabilir Ancak verilerin uygun işlemciye ulaştığından emin olmak için G/Ç dikkatli bir şekilde kontrol edilmelidir. Ayrıca, CPU'lar ayrı olduğundan, biri boşta otururken diğeri aşırı yüklenmiş olabilir ve bu da verimsizliklere neden olabilir. Bu formdaki çok işlemcili bir sistem, bellek gibi süreçlerin ve kaynakların çeşitli işlemciler arasında dinamik olarak paylaşılmasına izin verir ve işlemciler arasındaki farkı azaltabilir.





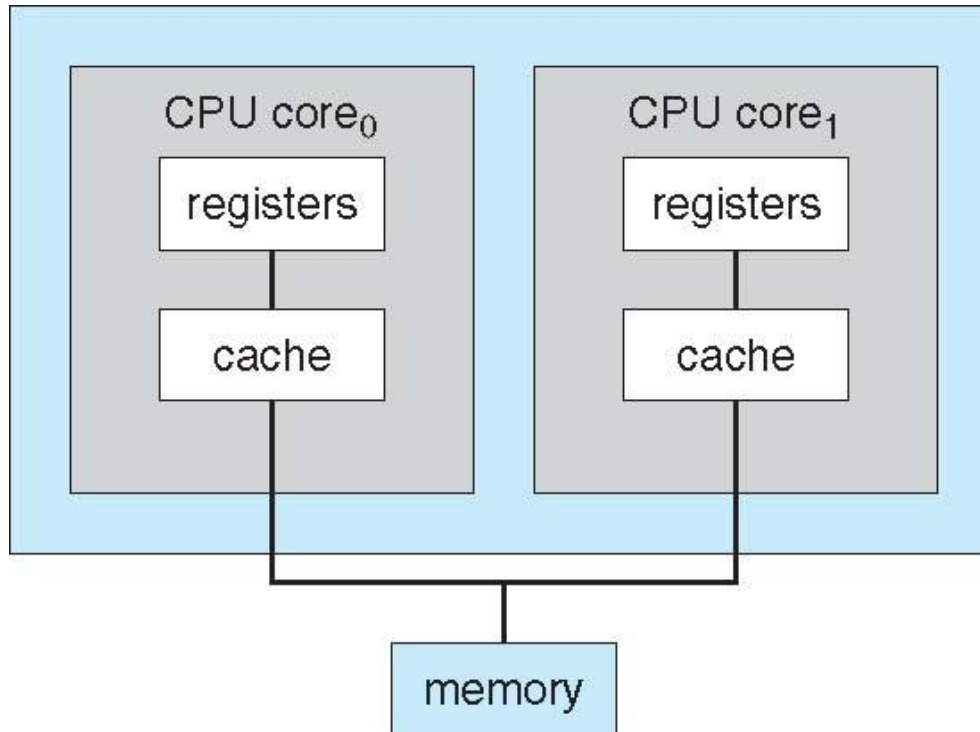
Simetrik Çok İşlemcili Mimari





Çift Çekirdekli Tasarım

- CPU tasarımında son zamanlardaki bir eğilim, tek bir çip üzerinde birden fazla bilgi işlem çekirdeğini dahil etmektir. Bu tür çok işlemcili sistemlere çok çekirdekli (**multicore**) denir. Daha verimli, çünkü veri alışverişi daha hızlı ve daha az elektrik gücü harcıyor.





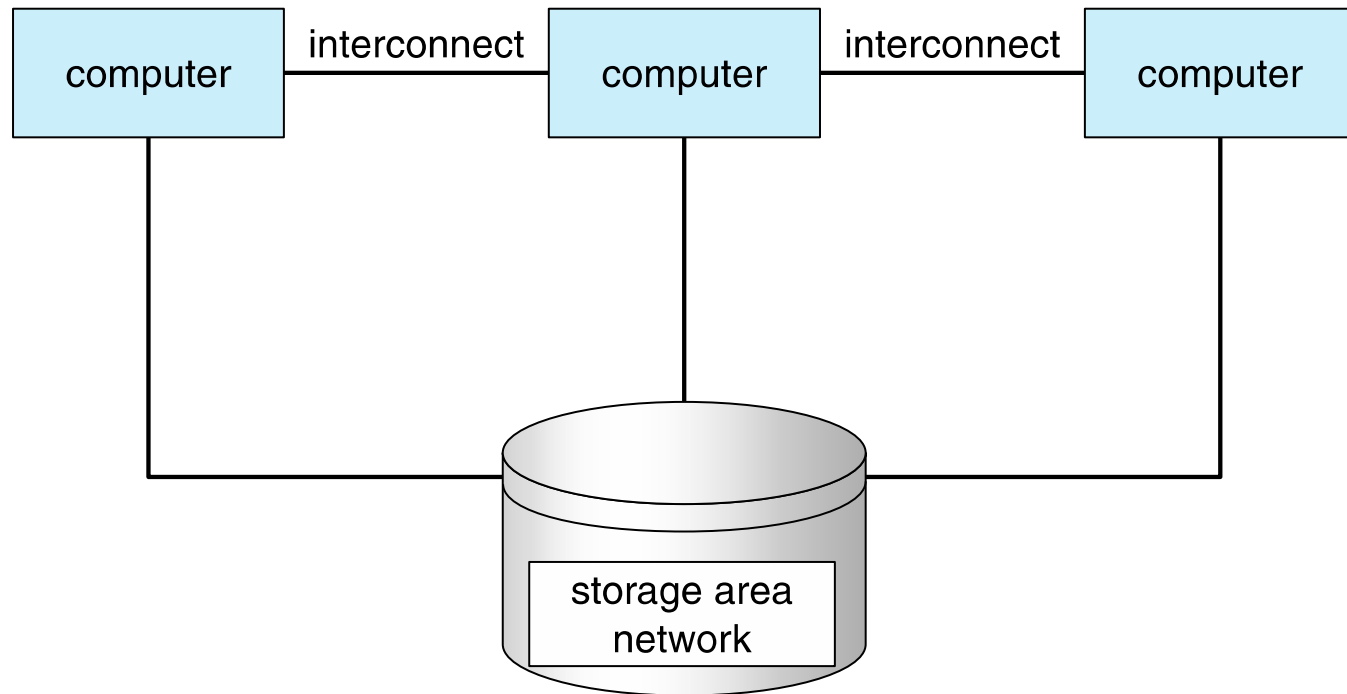
Kümelenmiş (Clustered) Sistemler

- Çok işlemcili sistemlere benzer fakat birçok sistem birlikte çalışır.
 - Genellikle depolama **storage-area network (SAN)** kullanılarak paylaşılır
 - Yüksek kullanılabilirliğe (**high-availability**) sahip bir servis imkanı sunar.
 - ▶ **Asimetrik kümeleme (Asymmetric clustering)** bir makine hot-standby mode'da bulunur. O gözlemler diğerleri çalışır.
 - ▶ **Simetrik kümeleme (Symmetric clustering)** hepsi uygulamaları çalıştırır ve birbirlerini izler.
 - Bazı kümeleme sistemleri yüksek performans hesaplama (**high-performance computing**) (**HPC**) için kullanılır.
 - ▶ Uygulamalar paralelleştirmeyi (**parallelization**) kullanacak şekilde yazılmalıdır.
 - Çakışmayı engelleme (**distributed lock manager** (**DLM**)) paralel kümeleme sistemlerinde kullanılır.





Kümelenmiş Sistemler





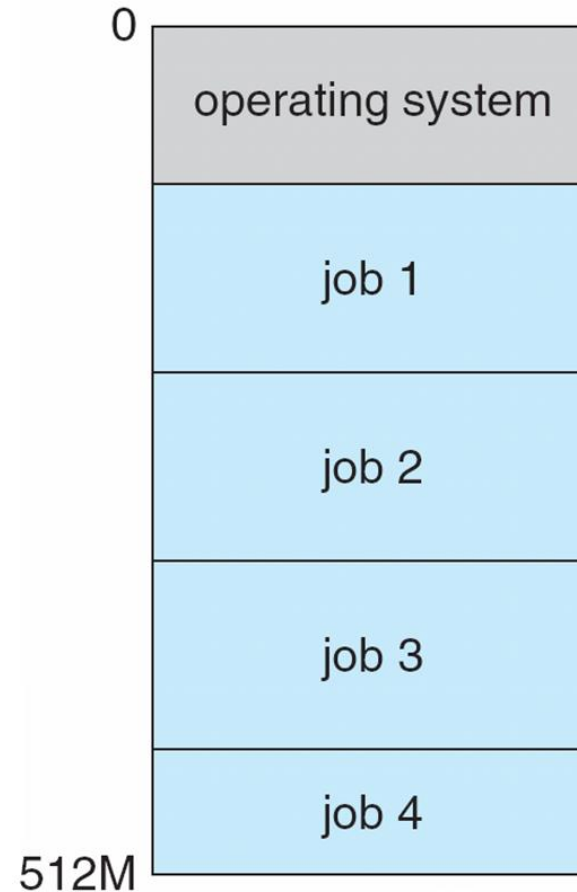
İşletim Sistemi Mimarisi

- **Çoklu programlama (Multiprogramming)** verimlilik için geçerlidir.
 - Tek kullanıcı CPU ve I/O cihazlarını her zaman meşgul edemez.
 - Çoklu programlama işleri (kod ve veri) düzenler, böylece CPU'nun her zaman bir tane işlemi yürütmesini sağlar.
 - Sistemdeki toplam işlerin bir alt kümesi bellekte tutulur.
 - Bir iş seçilir ve iş zamanlayıcısı (**job scheduling**) yoluyla çalıştırılır.
 - Beklemesi gerektiğinde (örneğin I/O için), OS başka bir işe geçer.
- **Zaman paylaşımı, Çoklu görev (Timesharing, multitasking)** CPU'nun sık sık iş değiştirdiği ve kullanıcıların çalışırken her bir iş ile etkileşimde bulunabildiği, interaktif hesaplama yöntemidir.
 - Yanıt süresi (**Response time**) <1 saniye olmalıdır.
 - Her kullanıcı hafızada en az bir program yürütür.
 - Aynı anda birden fazla iş çalışmaya hazırsa işlemci zamanlaması kullanılır (**CPU scheduling**).
 - Eğer işlemler hafızaya sığmazsa, değiş tokuş işlemi (**swapping**) yaparak veriler hafızaya alınır ve çıkartılır.
 - **Sanal hafıza (Virtual memory)** işlemlerin hafıza dışında da işletilmesini sağlar.





Çoklu Programlamalı Sistemlerde Hafıza Yerleşimi





İşletim Sistemi İşlemleri

- **Kesme Eğilimi (Interrupt driven)** (donanım ve yazılım tabanlı)
 - Donanım aygıtları tarafından kesme isteği gönderilebilir.
 - Yazılımsal kesme (beklenmeyen **(exception)** veya tuzak **(trap)**) :
 - ▶ Yazılım hataları (örnek, Sıfıra bölme sorunu)
 - ▶ İşletim sistemi servislerinden gelen istekler
 - ▶ Sonsuz döngü içeren, birbirini veya işletim sistemini değiştirmeye çalışan diğer işlem problemleri.





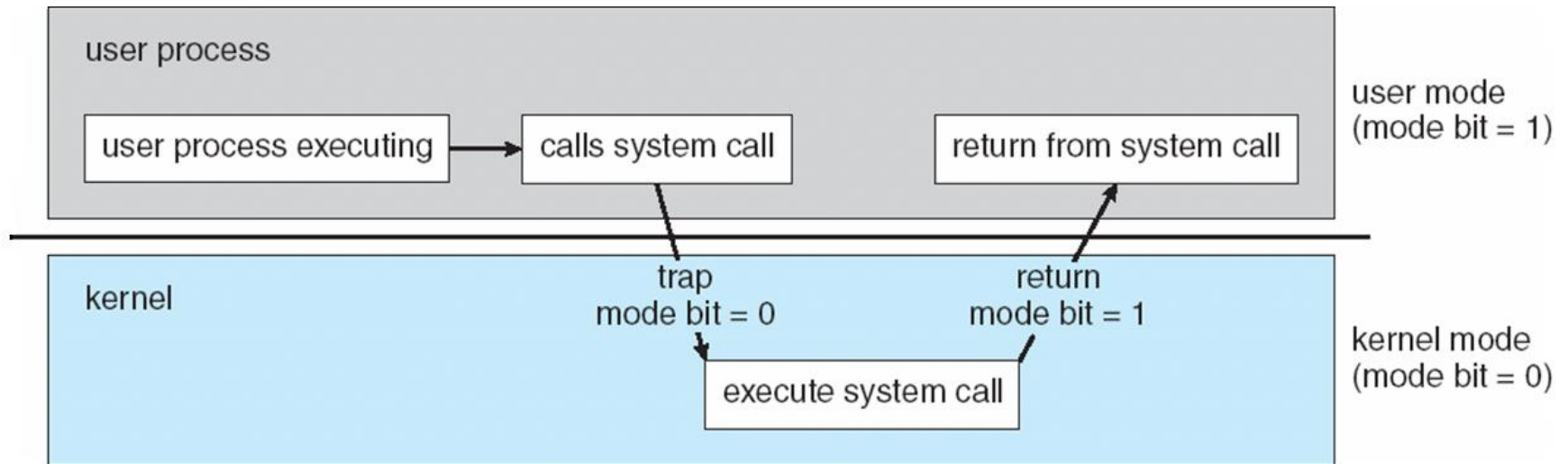
İşletim Sistemi İşlemleri (devamı)

- **Çift modlu (Dual-mode)** işleme işletim sistemini ve diğer sistem bileşenlerini korumayı sağlar.
 - **Kullanıcı modu (User mode)** ve **çekirdek modu (kernel mode)**
 - Donanım tarafından sağlanan mod biti (**Mode bit**)
 - ▶ Sistemin kullanıcı kodumu yoksa çekirdek kodumu çalıştırdığını ayırt etmekte kullanılır
 - ▶ Bazı komutlar ayrıcalıklı (**privileged instructions**) olarak tanımlanır ve çekirdek modunda çalıştırılır.
 - ▶ Sistem çağrısı geldiği zaman modu çekirdeğe çevirir ve sistem çağrısı bittiğinde tekrar kullanıcı moduna döner.
- Çoklu CPU'lar çok modlu işlemleri destekler
 - Örnek sanal makine yönetimi modu (**virtual machine manager (VMM)**) misafir sanal makineler için





Kullanıcıdan Çekirdek Moduna Geçiş





Zamanlayıcı

- Zamanlayıcı (Timer) sonsuz döngülere ve işlemci kilitlenmelerine engel olur.
 - Belirli bir zaman diliminde sonra kesme gönderebilir.
 - Fiziksel clock tarafından azaltılan bir sayaç tutulur.
 - İşletim sistemi sayacı ayarlar (ayrıcalıklı komut privileged instruction)
 - Sayaç sıfır olduğu zaman kesme oluşturulur.
 - İş zamanlayıcısı devreye girmeden önce sorun çıkaran program sonlandırılır ve kontrol tekrar ele geçirilir.





İşlem Yönetimi

- İşlem işletilen bir programdır. Sistem içerisinde bir iş birimidir. Program pasif bir varlıktır, süreç aktif bir varlıktır.
- Sürecin, görevini gerçekleştirmek için kaynaklara ihtiyacı vardır.
 - CPU, hafıza, I/O, dosyalar
 - Başlangıç verisi
- Süreç sonlandırma, yeniden kullanılabilir kaynakların geri alınmasını gerektirir.
- Tek iş parçacıklı işlem yürütmek için sonraki komutun konumunu belirten bir program sayacı
 - Süreç komutları, tamamlanana kadar sırayla, birer birer sıralı olarak yürütülür
- Çok iş parçacıklı işlemde (Multi-threaded process) iş parçacığı başına bir program sayacı vardır.
- Tipik bir sistem birçok işleme sahiptir, bunların bazıları kullanıcı bazıları işletim sistemi temellidir bir veya birkaç işlemcide eş zamanlı çalışırlar.





İşlem Yönetimi Faaliyetleri

İşletim sistemi, süreç yönetimi ile bağlantılı olarak aşağıdaki faaliyetlerden sorumludur:

- Hem kullanıcı hem de sistem süreçlerini oluşturma ve silme
- Askıya alma ve devam eden süreçler
- Süreç senkronizasyonu için mekanizmalar sağlamak
- Süreç iletişimi için mekanizmalar sağlamak
- Kilitlenme (deadlock) için mekanizmalar sağlama





Hafıza Yönetimi

- Bir programı yürütmek için komutların tümü (veya bir kısmı) bellekte olmalıdır.
- Programın ihtiyaç duyduğu verilerin tümü (veya bir kısmı) bellekte olmalıdır.
- Hafıza yönetimi hafızada nelerin olduğu:
 - CPU kullanımını ve bilgisayarın kullanıcılara dönüşünü optimize ederken kontrol eder.
- Hafıza yönetimi aktiviteleri:
 - Şu anda hangi bellek bölümlerinin kullanıldığını ve kime ait olduğunu takip etme
 - Hangi işlemlerin (veya işlem parçacıklarının) ve verilerin hafızaya girip çıkacağına karar vermek
 - Bellek alanını gerektiğinde ayırma veya geri iade etme





Dosya Sistemi Yönetimi

- Dosya yönetimi, bir işletim sisteminin en görünür bileşenlerinden biridir.
- Bilgisayarlar, birkaç farklı fiziksel ortam türü üzerinde bilgi depolayabilir.
- Manyetik disk, optik disk ve ssd en yaygın olanlarıdır.
- Bu ortamların her birinin kendine has özellikleri ve fiziksel organizasyonu vardır.
- Bu özellikler erişim hızı, kapasite, veri aktarım hızı ve erişim yöntemini (sıralı veya rastgele) içerir.
- Genellikle dosyalar programları (hem kaynak hem de nesne formları) ve verileri temsil eder.
- Veri dosyaları sayısal, alfabetik, alfanümerik veya ikili olabilir.
- Açıkçası, bir dosya kavramı son derece genel bir kavramdır.





Dosya Sistemi Yönetimi

■ Dosya Sistemi Yönetimi

- Dosyalar dizinler kullanılarak organize edilir
- Pek çok sistemde dizinlere veya dosyalara erişim kontrol edilmelidir: **erişim kontrolü (access control)**
- OS aktiviteleri
 - ▶ Dosya ve dizinlerin oluşturulması veya silinmesi
 - ▶ Dosyaların veya dizinlerin değiştirilmesi için mekanizmanın sağlanması
 - ▶ Dosyaların ikincil depolama birimi ile eşleştirilmesi
 - ▶ Dosyaların kalıcı depolama birimlerine yedeklenmesi





Disk Yönetimi

- Genellikle diskler, hafızaya sığmayan verileri ya da uzun süre tutulacak verileri tutmakta kullanılır
- Bilgisayar işleminin tüm hızı, disk alt sistemi ve algoritmalarına dayanır
- OS aktiviteleri
 - Boş alan yönetimi
 - Depolama alanı ayrımı
 - Disk zamanlama (scheduling)
- Bazı depolama birimlerinin hızlı olması gerekmez
 - CD, DVD, Manyetik teypler
 - Genelde OS veya uygulamaları tarafından yönetilmelidir.
 - WORM (write-once, read-many-times) ve RW (read-write) erişim modlarında çalışabilirler.





Depolama Birimlerinin Performansı

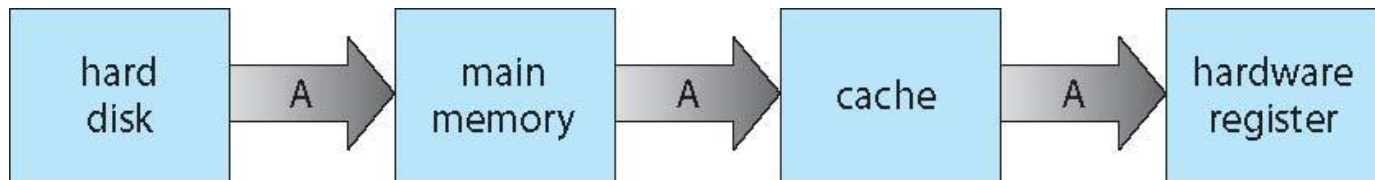
Level	1	2	3	4	5
Name	registers	cache	main memory	solid state disk	magnetic disk
Typical size	< 1 KB	< 16MB	< 64GB	< 1 TB	< 10 TB
Implementation technology	custom memory with multiple ports CMOS	on-chip or off-chip CMOS SRAM	CMOS SRAM	flash memory	magnetic disk
Access time (ns)	0.25 - 0.5	0.5 - 25	80 - 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





Diskten Registera “A” Verisinin Aktarımı

- Çok işlemli ortamlar, en güncel değeri kullanmak konusunda dikkatli olmalıdır (depolama hiyerarşisinin neresinde tutuluyorsa tutulsun)



- Çok işlemcili sistemlerde **ön bellek tutarlılığı (cache coherency)** donanım seviyesinde sağlanmalı ve tüm işlemciler en güncel değere sahip olmalıdır.
- Dağıtık ortamlarda durum daha da karmaşıktır
 - Verinin birden fazla kopyası olabilir





I/O Alt Sistemi

- İşletim sisteminin amaçlarından biri donanım cihazlarının karmaşıklıklarını kullanıcıdan gizlemektir
- I/O alt sisteminin sorumlulukları:
 - **Tampon bellek işlemleri (buffering)** – veriyi bir yerden diğer yere aktarırken geçici olarak saklamak
 - **Ön bellek işlemleri (caching)** – veriyi geçici olarak daha hızlı depolama birimine aktarmak
 - **Kuyruklama (spooling)** – bir işin çıktısını diğer işin girdisi haline getirmek
 - Genel aygıt sürücüsü arayüzü
 - Belirli donanım cihazları için sürücüler





Koruma ve Güvenlik

- Koruma (Protection) - Süreçlerin veya kullanıcıların OS tarafından tanımlanan kaynaklara erişimini kontrol eden herhangi bir mekanizma
- Güvenlik (Security)- sistemin iç ve dış saldırılara karşı savunulması
 - Hizmet reddi, solucanlar, virüsler, kimlik hırsızlığı, hizmet hırsızlığı da dahil olmak üzere geniş bir aralık
 - Sistemler öncelikle kullanıcıları, kimin ne yapabileceğine göre sınıflandırır
 - Kullanıcı kimlikleri (kullanıcı kimlikleri, güvenlik kimlikleri) (**user IDs, security IDs**), kullanıcı başına bir isim ve birleşik numara içerir.
 - Kullanıcı adı daha sonra, erişim kontrolü amacıyla, kullanıcının sahip olduğu tüm dosya ve işlemlerle ilişkilendirilir.
 - Grup adı (**group ID**) da benzer şekilde bir grup kullanıcıyı belli işlem ve dosyalarla ilişkilendirmek ve erişim kontrolü sağlamak amacıyla kullanılır
 - Ayrıcalık yükseltme (**Privilege escalation**), kullanıcının daha fazla hakla etkin kimliğe geçmesine izin verir.

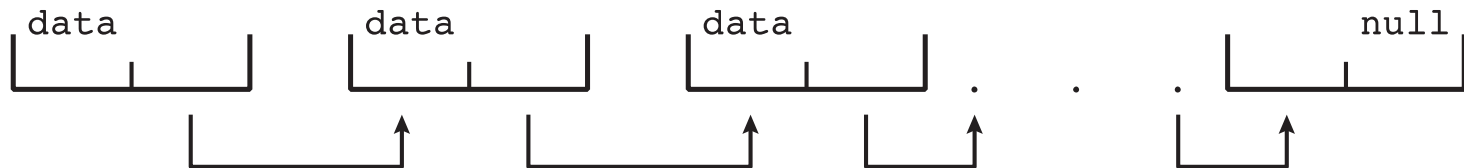




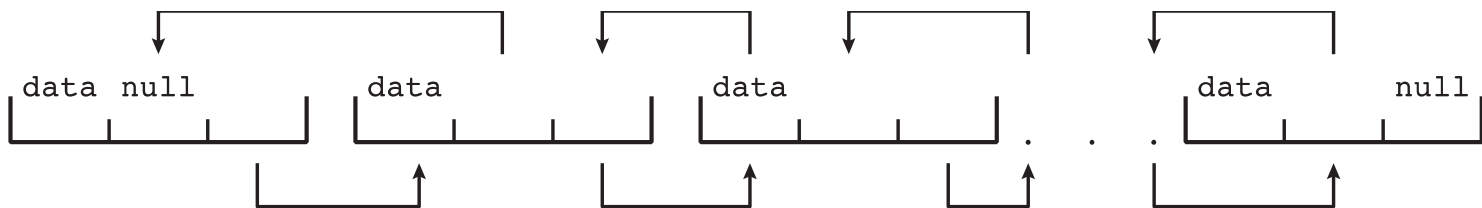
Çekirdek Veri Yapıları

- Standart programlama veri yapılarına çok benzer

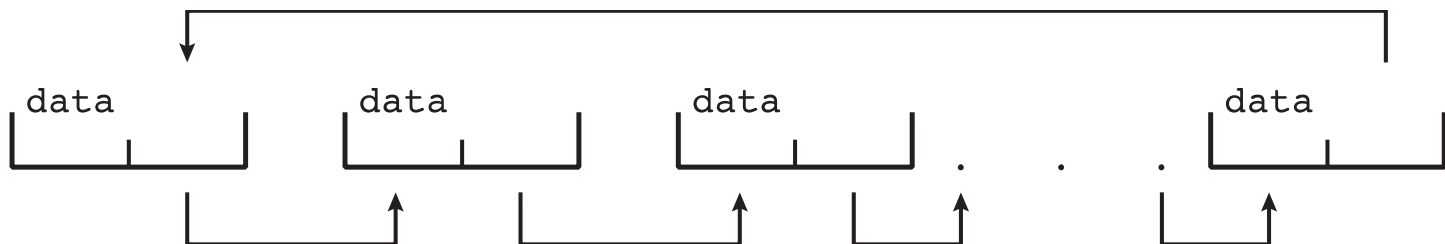
- ***Singly linked list***



- ***Doubly linked list***



- ***Circular linked list***



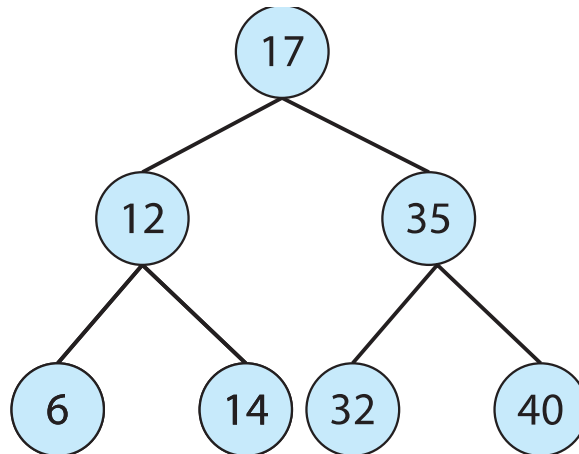


Çekirdek Veri Yapıları

■ Binary search tree

left \leq right

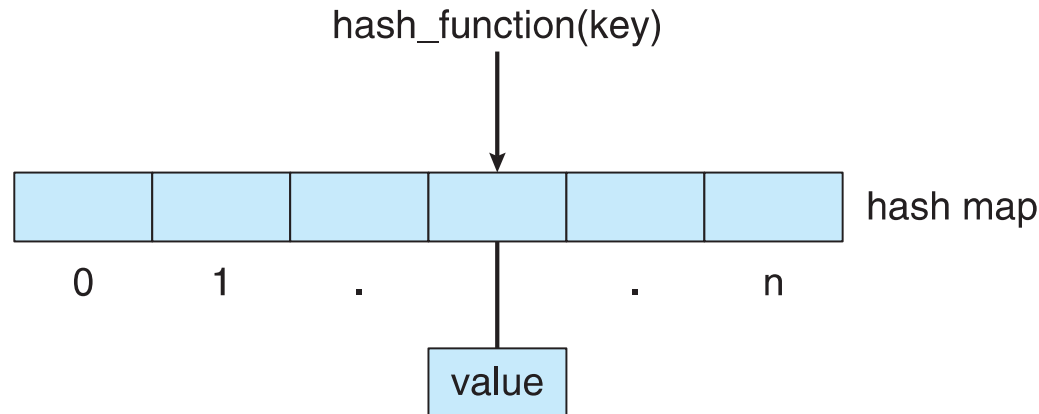
- Arama performansı $O(n)$
- **Dengelenmiş binary search tree** $O(\lg n)$





Kernel Data Structures

- **Hash function** can create a **hash map**



- **Bitmap** – string of n binary digits representing the status of n items
- Linux data structures defined in
include files `<linux/list.h>`, `<linux/kfifo.h>`,
`<linux/rbtree.h>`





Bilgisayar Ortamları

- Bağımsız genel amaçlı makineler
- Ancak çoğu sistem diğerleriyle bağlantı kurduğunda işler biraz daha karmaşıklaşır. (örnek, internet)
- Portallar (**Portals**) dahili sistemlere web erişimi sağlar
- Ağ bilgisayarları **Network computers** (**thin clients**) Web terminalleri gibidir.
- Mobil bilgisayarlar kablosuz ağlarla birbirine bağlanır.
- Ağ oluşturma olağan bir durum – ev bilgisayarları bile internet saldırılarından korunmak için güvenlik duvarı (**firewall**) kullanır.





Bilgisayar Ortamları- Mobil

- Cep telefonları, tabletler
- Geleneksel bilgisayarlar ile onlar arasındaki fark nedir?
- Ekstra özellik - daha fazla işletim sistemi özellikleri (GPS, jiroskop)
- Artırılmış gerçeklik (***augmented reality***) gibi yeni uygulama türlerine izin verir
- Bağlantı için IEEE 802.11 kablosuz veya hücresel veri ağlarını kullanır
- Liderler **Apple iOS** ve **Google Android**





Hesaplama Ortamları – Dağılık

- Dağılık hesaplama
 - Birbirine bağlı ayrı, muhtemelen heterojen sistemlerin toplanması
 - ▶ **Network** ortak iletişim yolu, **TCP/IP** en yaygın protokol
 - **Local Area Network (LAN)**
 - **Wide Area Network (WAN)**
 - **Metropolitan Area Network (MAN)**
 - **Personal Area Network (PAN)**
 - **Network Operating System** ağdaki sistemler arasında belli özellikler sağlar.
 - ▶ İletişim şeması, sistemlerin mesaj alışverişi yapmasına izin verir

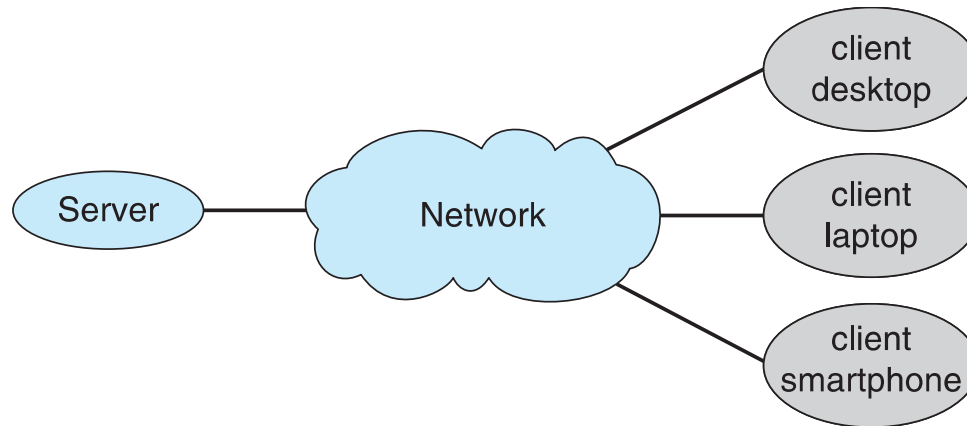


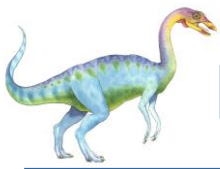


Hesaplama Ortamları – İstemci-Sunucu

■ İstemci-Sunucu Hesaplama

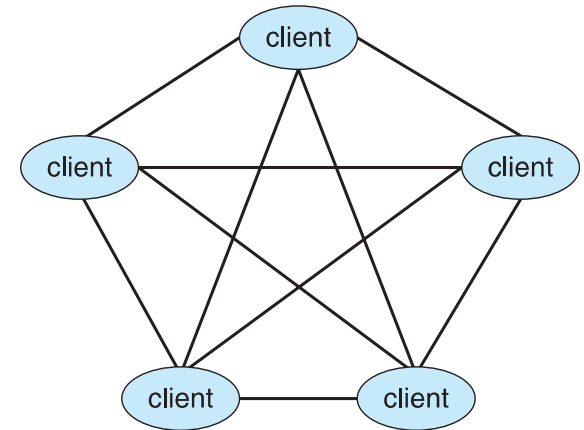
- Akıllı PC'ler tarafından desteklenen aptal terminaller
- Şu an pek çok sistem sunucu (server) olarak kullanılıyor, ve istemcilerin (clients) isteklerine cevap veriyor.
 - ▶ **Hesaplama-sunucu sistemi** istemcilere çeşitli servisler sağlayan bir arayüz sunar (örn. veritabanı)
 - ▶ **Dosya sunucu sistemi** istemcilere dosyaları kaydetmeyi ve indirmeyi sağlayan bir arayüz sunar





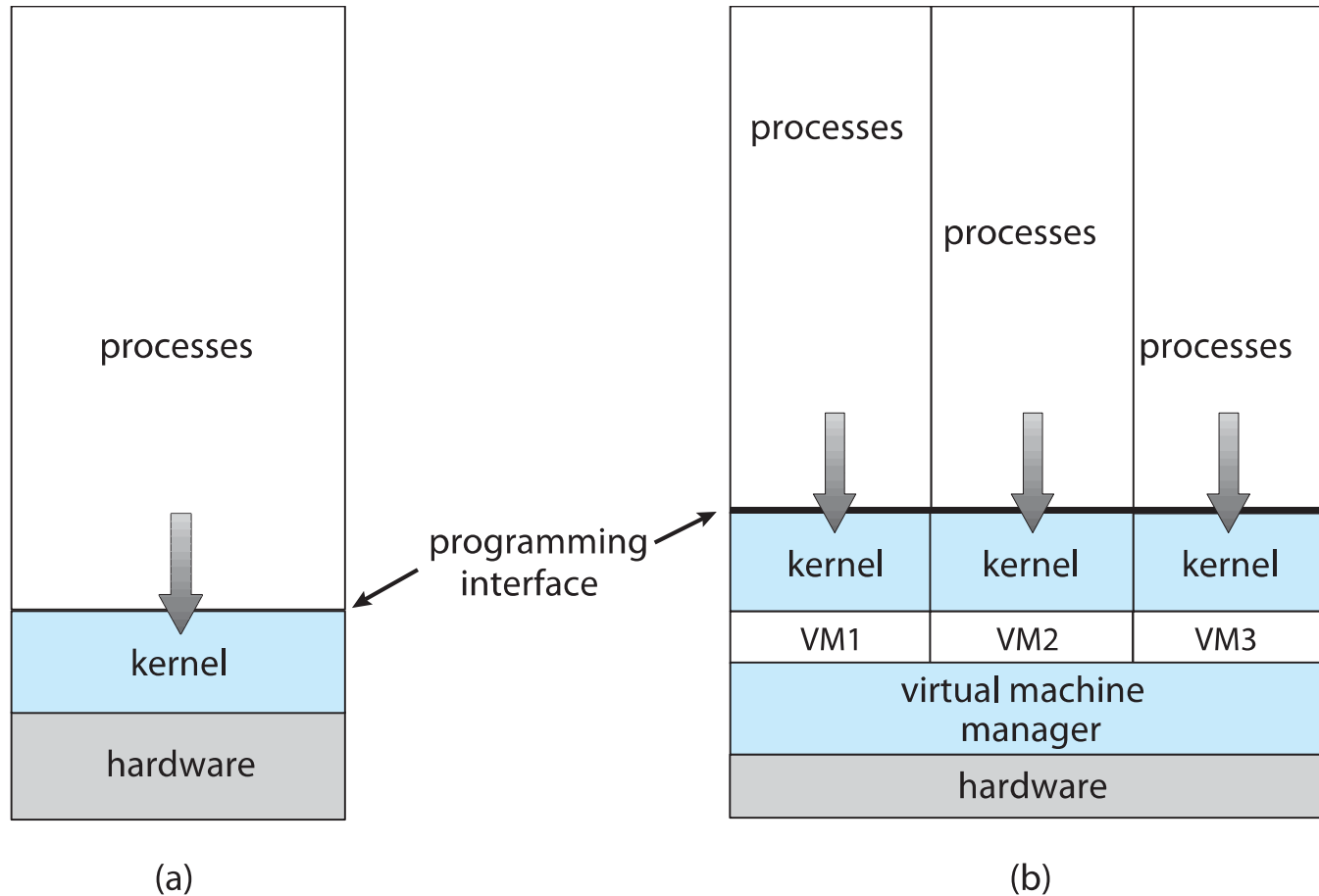
Hesaplama Ortamları- Uçtan Uca Sistemler

- Dağıtık sistemlerin bir başka örneği
- P2P istemci ve sunucu arasında ayırım yapmaz.
 - Her bir sistem bir uç olarak ele alınır
 - Her bir uç hem istemci hemde sunucu gibi davranabilir
 - Uçlar öncelikle P2P ağa bağlanmalıdır
 - ▶ Kendini bu ağdaki merkezi kayıt sistemine kaydetmelidir, veya
 - ▶ **Keşif protokolü (discovery protocol)** ile istekte bulunmalı veya daha önce bulunulan istekleri karşılamalıdır
 - Örnek, **Voice over IP (VoIP)** Skype gibi





Hesaplama Ortamları- Sanallaştırma

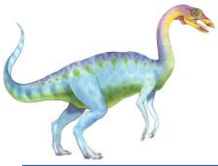




Hesaplama Ortamları – Bulut bilişim

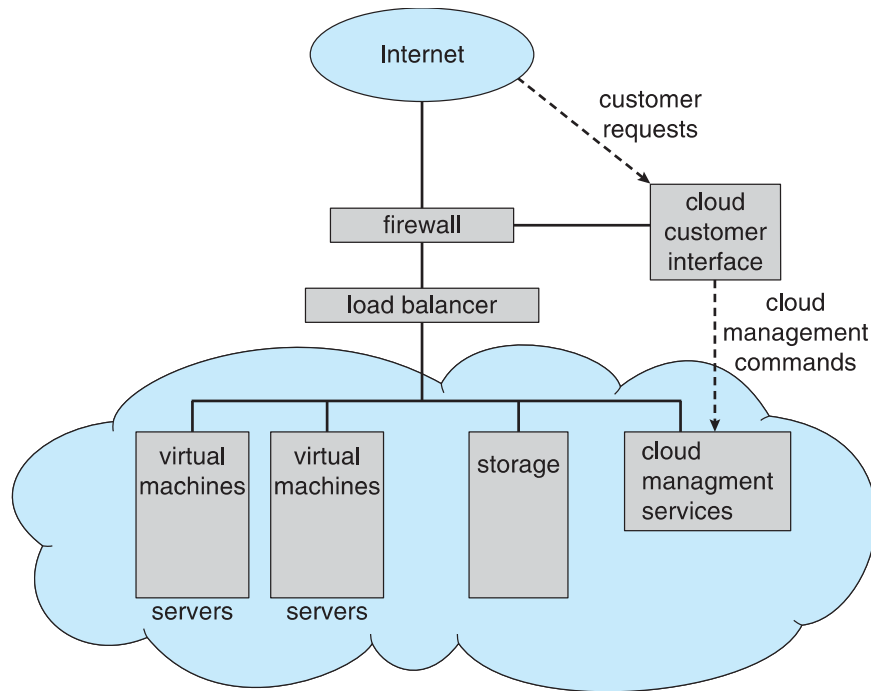
- Ağ üzerinden bir bilgisayar olarak bilgi işlem, depolama ve hatta uygulamalar sunar
- Sanallaştırmanın mantıksal uzantısı, çünkü sanallaştırmayı işlevsellik tabanı olarak kullanır.
 - Amazon EC2, binlerce sunucuya, milyonlarca sanal makineye, internete bağlı petabaytlarca depolama alanına sahiptir.
 - **Public cloud** – ücretini yatıran herkes için
 - **Private cloud** – şirket için
 - **Hybrid cloud** – genel ve özel kullanımlar için
 - Software as a Service (**SaaS**) – İnternet üzerinden kullanılabilen bir veya daha fazla uygulama (i.e., kelime işlemci)
 - Platform as a Service (**PaaS**) – İnternet üzerinden uygulama kullanımı için hazır yazılım (i.e., veri tabanı sunucusu)
 - Infrastructure as a Service (**IaaS**) – İnternet üzerinden kullanılabilen sunucular veya depolama (i.e., yedekleme için depolama alanı)





Hesaplama Ortamları – Bulut bilişim

- Bulut bilişim ortamları, geleneksel işletim sistemleri, artı VMM'ler ve bulut yönetim araçlarından oluşur.
 - İnternet bağlantısı güvenlik duvarları gibi güvenlik gerektirir
 - Yük dengeleyiciler, birden fazla uygulamaya yayılmış trafiği yaydılar





Hesaplama Ortamları – Gerçek Zamanlı Gömülü Sistemler

- Yaygın biçimde kullanılırlar
 - Özel amacı olan işletim sistemleri





Açık kaynak İşletim Sistemleri

- Bu işletim sistemleri, **kapalı kaynak kodundan (closed-source)** ziyade **kaynak kod (source-code)** formatında sunuluyor
- Kopya koruma **copy protection** ve Dijital Hak Yönetimi **Digital Rights Management (DRM)** hareketine karşı bir harekettir.
- **Free Software Foundation (FSF)** tarafından başlatıldı, which “copyleft” diye tabir edilen **GNU Public License (GPL)**
- **GNU/Linux** ve **BSD UNIX** örnek verilebilir. **Mac OS X** ve birçok işletim sisteminin çekirdeği



Bölüm 1 Sonu

