

# BLM3580

# Sistem Programlama

---

2021-2022 GÜZ YARIYILI

DR.ÖĞR.ÜYESİ GÖKSEL BIRICIK

# Sistem Programlama

---

# Sistem Programlama

---

Sistemin sözlük anlamı: ***bir sonucu elde etmeye yarayan yöntemler düzeni***

## **Sistem programlamanın hedefleri:**

- Kullanılacak bilgisayar donanımının kullanıcı isteklerine cevap verecek sistem ve uygulama yazılımları ile donatılması
- Arzu edilen veya beklenen başarımda çalışması
- Yapının sürekliliği
- Olası gelişmelere uyumu

## **İçeriği oluşturan bileşenlerin temelleri:**

- Bilgisayar organizasyonu
- Programlama dilleri
- Derleyici tasarımı
- İşletim sistemleri

# Sistem Yazılımları

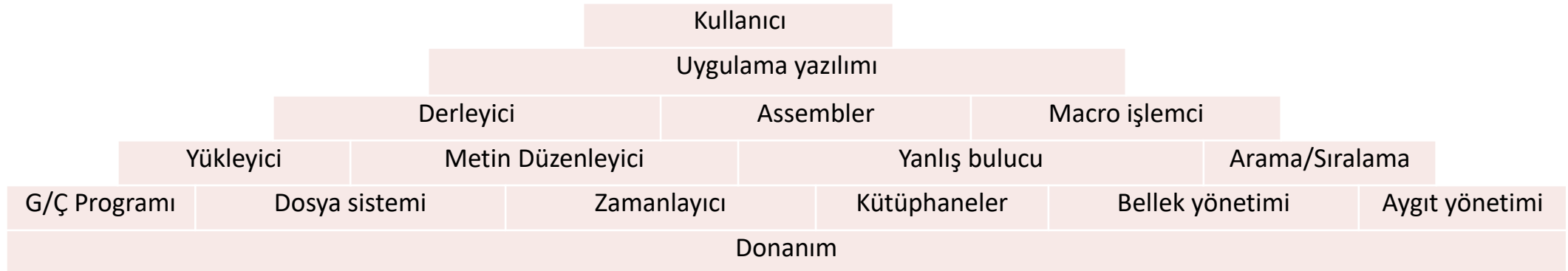
İlk başlarda, çok basit işlemler ancak bazı temas noktalarının açılması veya kapatılması (hardwired programming) sayesinde verilen komutlar ile yapılabiliyordu.

İhtiyaçlar ve beklentiler arttı. Bilgisayarların karmaşık işlemleri yapmasını sağlayacak sistem yazılımları olan derleyici (compiler), makro işlemci (macro processor), yükleyici (loader), yanlış bulucu (debugger) ve işletim sistemi (operating system) yazılımları türedi .

Sistem yazılımları ihtiyaca göre geliştirilmiş uygulama programları ile donanım arasında bir ara katman olarak yer alır ve sistemin genel başarımında en önemli rolü oynarlar.

Yazılım	Uygulama Yazılımları	Programlama dilleri kullanılarak kullanıcı isteklerini cevaplamak üzere geliştirilmiş yazılımlar.
	Sistem Yazılımları	Derleyici, Yükleyici, Macro İşlemci, İşletim Sistemi
Donanım		

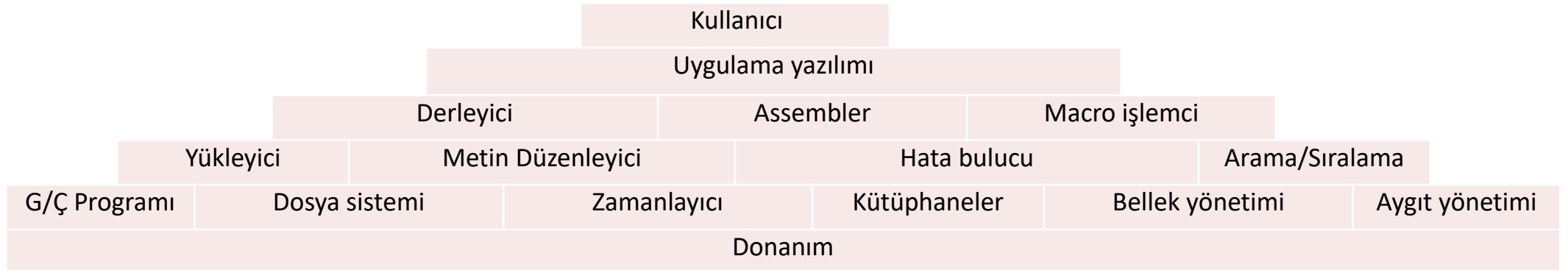
# Sistem Programlama Piramit Modeli-1



Donanım üzerinde işletim sistemi.

- Daha üst katlardan gelen isteklere uygun olarak işlemci, bellek, çevre aygıtlar ve bilgi olarak özetleyebileceğimiz kaynak veya servisleri tahsis eder, kullanır ve ihtiyacın bitimini takiben geri alır
- Giriş/çıkış işlemleri, dosya sistemi, zamanlayıcı (scheduler), bellek yönetim ve aygıt yönetim modülleri ile donatılmıştır

# Sistem Programlama Piramit Modeli-2



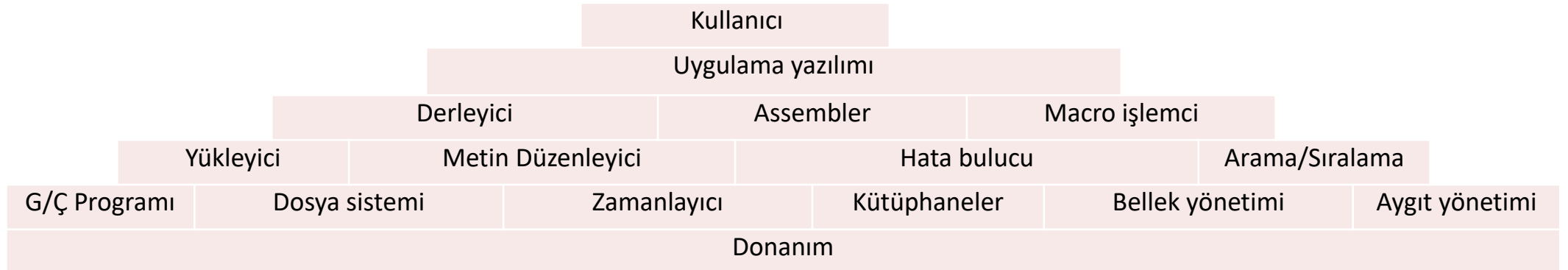
Yükleyici, derleyiciden bilgisayarın anladığı makine kodu (object code) olarak elde edilen programın belleğe yerleştirilmesi ve çalıştırılma için gerekli hazırlıkların yapılmasından sorumlu.

Programın belleğe yüklenmesinden sonra doğru çalışıp çalışmadığını, var ise yanlışların nerede olduğunu bulmak için hata bulucu (debugger) programlardan yararlanır.

Sistem üzerinde bir takım bilgilere erişebilmek için arama ve sıralama,

Bir dosya üzerinde değişiklik yapmak gerektiğinde metin düzenleyici kullanılır.

# Sistem Programlama Piramit Modeli-3

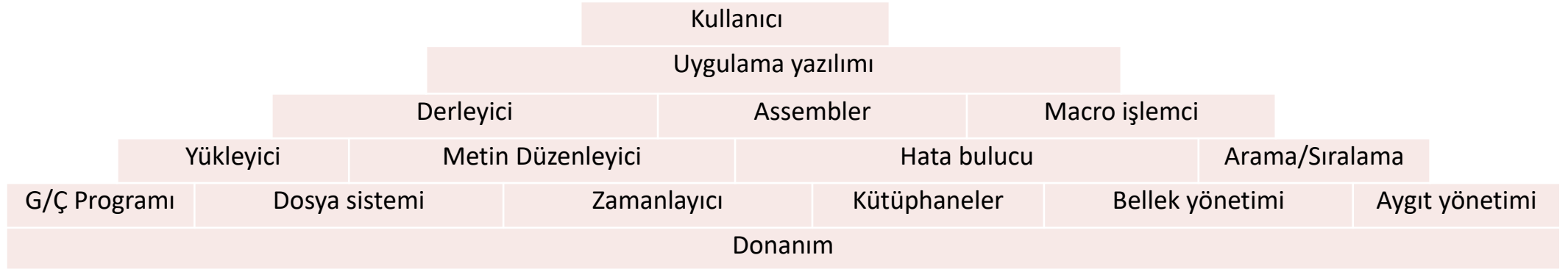


Derleyici, kullanıcı tarafından yüksek seviyeli sembolik bir programlama dili yardımıyla yazılmış bir programın donanım tarafından anlaşılan makine koduna çevirmek için kullanılır

Assembler ise assembly dili derleyicisidir. Assembly dilinde yazılmış veya derleyici tarafından assembly sembolik komutlarına dönüştürülmüş programlar üzerinde işlem yaparak donanımın anlayacağı makine kodlarını üretir.

Macro işlemci, programlar içinde kullanılan makroların gerek assembler gerek ise derleyiciler tarafından işlemine dâhil edilmesini sağlamakla görevlidir.

# Sistem Programlama Piramit Modeli-4



Uygulama yazılımları, kullanıcının ihtiyacını karşılamak üzere hazırlanmış, donanımın sağladığı imkânları kullanarak sonuç üreten programlardır.

Elektronik tablo işlem, kelime işlem, sunum, grafik tasarım, muhasebe gibi konularda hazırlanmış programlar olup, son kullanıcının birebir temasta olduğu grafik ara yüzler ile sistemi amacına uygun işlemleri yapmak üzere kullanmasına olanak verirler.



# Bilgisayar Sistemlerinin Sınıflandırılması

---

Fiziksel büyüklük, hız, bellek, diğer özellikler ile sınıflandırılabilir.

- Süper Bilgisayar (Supercomputer)
- Süper Mini Bilgisayar (Minisupercomputer)
- Ana Bilgisayar (Mainframe Computer)
- Kurumsal Sunucu (Enterprise Server)
- Orta Boy Bilgisayar (Mini-Computer)
- İş İstasyonu (Workstation)
- Kişisel Bilgisayar – Mikro Bilgisayar (Personal Computer – PC, Desktop Computer –Microcomputer)
- Dizüstü Bilgisayar (Laptop Computer)
- Kişisel Sayısal Asistan (Personal Digital Assistant - PDA, Smart Device)
- Giyilebilir Bilgisayar (Wearable computer)

# Bilgisayar Sistemi Çeşitleri

---

## Merkezi Sistemler

## Dağıtılmış Sistemler

- Sıkı Eşli
- Gevşek Eşli

## Sanallaştırılmış Sistemler

- Donanım Seviyesi
- Yazılım Seviyesi
  - Hypervisor Mimari
  - Konaklı Mimari
- Uygulama Seviyesi

## Bilgisayar Ağları

# Merkezi Bilgisayar Sistemleri

---

1970'lerin sonlarına kadar: merkezi yapıda çalışan, özel eğitilmiş personel, özel şartların sağlandığı (*yükseltilmiş taban, sabit sıcaklık, düzgün besleme gerilimi, tozdan arıtılmış*) ortamlarda kullanılan, sınırlı sayıda firma tarafından üretilen cihazlar

Tüm öğeler (*donanım, sistem ve uygulama yazılımları*) donanımı sağlayan firma tarafından sağlanırdı.

Bilgisayar ihtiyacınız varsa firmayı seçersiniz, firma sizin için gerekli bütün öğeleri bir araya getirir ve anahtar teslimi şeklinde sistemleri kurar böylece alıcıyı detaylardan soyutlardı.

- Hem donanım, hem de yazılımlar üretici firmanın patentli ürünü, 3.kişilere kapalı. Problemin tüm sorumluluğu sağlayıcındı.

“Uygulamalarımızı nasıl geliştireceğiz?” üretici firma uygulama geliştirmeniz için size bu amaçla geliştirdiği ürünü (programlama dili ve bu dil ile uygulama geliştirmek için gerekli araçlar-derleyicileri) önermek ve satmaktan mutlu olurdu.

Ya yanlış seçim yaptıysak?

Günümüzde dünya sıralamasındaki ilk 25 banka faaliyetlerini mainframe bilgisayar sistemleri üzerinde sürdürmekte, Fortune 500 sıralamasındaki firmaların %71'i IBM System Z müşterisi ve dünya çapında hayat/sağlık sigortası sağlayan firmalardan 9/10'u yüksek hacimli işlemlerinde mainframe bilgisayar sistemlerini tercih etmektedir.

# Dağıtılmış Sistemler

---

1980'lerde kişisel bilgisayarlar doğdu.

1990'larda büyük bir hız ile artarak çeşitlendi. Üretici sayısı arttı, açık sistemler ve standartlar kapalıların yerini aldı. Kullanıcıların istedikleri özelliklerdeki bilgisayar sistemlerini seçme özgürlükleri doğdu.

- Açık sistemler, birlikte çalışabilir (inter-operable), taşınabilir (portable), yönetilebilir (manageable) ve ölçeklenebilir (scalable) oldukları için fayda yüksek.
- Çok seçenek arasından hangisinin doğru teknolojik yatırım olduğuna karar vermek, mevcut ürünler arasında hangilerinin işe uygun olduğunu belirlemek, sistemi oluşturan parçaların birbirleri ile uyumlu çalışıp çalışmayacağını tespiti ve oluşturulan yapının ne derece esnek olduğunun belirlenmesi çok zor.
- Lokantada menüden sipariş vermek gibi. Yaptığınız seçimler sonunda elde edeceğiniz zevk tamamen sizin seçimlerinizin uyumuna bağlıdır.
- Oluşturulan sistemin çalıştırılması doğrudan oluşturanın veya kullananın sorumluluğundadır.
- Paket çözümler satan veya danışmalık yapan firmalar sizin probleminizi çözmek için bulunacaktır. Ancak bu servisin de maddi bir karşılığı olacağı unutulmamalıdır.

# Dağıtılmış Sistemler

---

Merkezi sistemlerden farklı olarak ihtiyaç duyulan hesaplama gücüne sahip olabilmek için birden fazla işlem ünitesinin (Multi processor) farklı şekillerde bir araya gelmesi ile oluşurlar

- İşlem birimleri arası mesafe  $< 1\text{cm}$  ise SMP (Symetric Multiprocessor-Massively Parallel),  $< 1\text{m}$  ise Multi-processor,  $> 10\text{m}$  ise Muti-computer (Computer Network) olarak da adlandırılabilir.

## Sıkı Eşli Sistemler (Tightly Coupled)

- Belli bir veri yolu (bus) seviyesinde birbirleri ile bağlı olan, kendi aralarındaki haberleşmenin ortak bellek (shared memory) ile sağlandığı sistemler.
- genelde bir baskılı devre üzerinde yer alan birden fazla merkezi işlem birimlerinden oluşan, kendi aralarındaki haberleşmenin yoğun ve ortak bellek üzerinden yürütüldüğü, hızın yüksek, birimlerin birbirlerine bağımlılığının yüksek (koordinasyonun yüksek) olduğu, nispeten pahalı sistemler

## Gevşek Eşli Sistemler (Loosely Coupled)

- Her biri kendi başına çalışabilen (özerk-otonom), aralarındaki haberleşmenin hızlı bir veri ağı üzerinden yollanan mesajlar ile sağlandığı sistemler.
- birimler arasındaki haberleşme yoğunluğu ve birimlerin birbirleri ile olan bağımlılıklarının düşük, haberleşmenin yavaş, olduğu nispeten ucuz sistemler.

# Sanallaştırılmış Sistemler

---

Tek bir fiziksel sunucunun, özel bir sanallaştırma yazılımı yardımı ile çok sayıda, birbirinden bağımsız, kendi işletim sistemini ve uygulama yazılımlarını çalıştırabilen sanal sistemler olarak kullanılmasına imkân veren teknoloji

İlk olarak mainframe'lerde, yüksek hesaplama gücündeki işlemci ve buna bağlı olan giriş/çıkış birimlerini farklı işleri veya yükleri çalıştıracak şekilde parçalara (partition) ayrılması ile gerçekleştirilmiştir. satıcı bağımlı olan donanım ve yazılımların kullanılması gereklidir.

IBM ticari olarak 1970'li yıllarda, S-370 sistemi üzerinde geliştirdiği VM/370 işletim sistemiyle sanallaştırmayı uygulamış, günümüzde de donanımları üzerinde ZVM işletim sistemi ile bu teknolojiyi kullanmaya devam etmektedir.

Günümüzde x86 platformunda, gelişen işlemci teknolojileri ve sanallaştırma yazılımları (hypervisor) ağırlıklarını arttırmış, endüstri standardı parçalar olarak hemen hemen her üreticinin sunucu sisteminde hizmet verebilir hale gelmiştir.

- Vmware (Vmware Server / Vmware Workstation), Citrix (XenServer), Oracle (Virtual Box), Microsoft (HyperV Server / Virtual PC 2007) ve Red Hat firmaları tarafından gerek sunucu taraflı (bare metal-Type1) gerek ise son kullanıcı taraflı (hosted-Type-2)

# Sanallařtırma Tercih Sebepleri

---

Sunucu Birleřtirme (Server Consolidation)

Masaüstü Sanallařtırma (Desktop Virtualization)

Depolama Birimlerini Birleřtirme (Storage Consolidation)

Yazılım Tanımlı Veri Merkezi (Software-Defined Data Center)

# Sanallaştırmanın Hedefleri

---

1. Yüksek yararlanabilirlik (Availability)
  1. Hızlı saklama, kopyalama ve tedarik (provisioning) özellikleri sayesinde sıfır aksaklık süresi (zero-downtime)
  2. Canlı bakım, onarım desteği
  3. Arıza (Fault) toleransının arttırılması
2. Felaketten kurtarma (Disaster recovery)
3. Yük dengeleme (Load balancing)
4. Veri merkezlerinin etkinliğini arttırmak
  1. Kaynakları optimize ederek iş yüklerinin dengelenmesi, otomatik olarak kaynak durumlarının sorgulanması ve farklı sanallaştırma katmanlarının yaratılabilmesi
  2. Düşük işletme ve bakım maliyeti (Sistemlerin sayısal olarak azalmasının bir sonucu olarak kapladıkları alanın azalması, fiziksel mekân maliyetini de azalacaktır. Benzer şekilde sistemlerin ürettikleri toplam ısıнын azalması bunun soğutulması için harcanan enerjiden de tasarruf sağlamaya yardımcı olacaktır. Ayrıca daha az sayıda sistem görelisi olarak daha az sayıda personel tarafından yönetilebilir.)
5. Kolay ölçeklenebilirlik (Scalability)



# Sanallaştırma ve Yararlanabilirlik

---

Özellikle veri merkezlerinin (data center) yararlanabilirliğini belirlemede Tier kavramından yararlanılmaktadır. Tier kavramı, veri merkezinin çalışırılık süresini (up-time) belirlemede kullanılan standart yöntem bilimi (methodology) ifade eder. Bu amaçla;

Tier 1 = Tek sunucu ve tek üst bağlantılı (uplink) (Non-redundant) sistemleri

Tier 2 = Tier 1 + Gereğinden fazla (Redundant) kapasiteli sistemleri

Tier 3 = Tier 1 + Tier 2 + Çift güç kaynaklı ve çok kanallı üst bağlantılı (uplinks) sistemleri

Tier 4 = Tier 1 + Tier 2 + Tier 3 + Tüm unsurları (üst bağlantılar, depolama, soğutma/havalandırma, ağ ve sunucu sistemleri) hatadan etkilenmez (fully fault-tolerant) ve çift güç bağlantılı sistemleri tarif eder.

# Sanallaştırma ve Yararlanabilirlik

---

IDC tarafından yapılan bir sınıflamada ise;

AL 1 = Disk aynalama (mirroring) veya RAID (Veri bütünlüğünün korunması)

AL 2 = Kullanıcı yükünü kaydırma (Kullanıcı tarafından fark edilebilir kesinti)

AL 3 = Otomatik olarak yükü üzerine alma (fail over) (Kullanıcı tarafından zorlukla fark edilebilir kesinti)

AL 4 = Hatadan etkilenmez (fault-tolerant) şeklinde ifade etmektedir.

Her ne şekilde olursa olsun, temelde sistemlerde ne kadar bir çalışmama süresinin (down-time) kabul edilebileceğidir. Bunun hesabında; A değişkeni taahhüt edilen sürekli çalışma saat değerini (Committed hours of availability), B değişkeni ise devre dışı olunan saat değerini göstermek üzere hedeflenen yararlanılabilirlik değeri  $((A-B)/A)*100\%$  formülü ile hesap edilir. %90 değeri 36,5 gün/yıl, %99 değeri 3.65 gün/yıl, %99.99 değeri 50 dakika/yıl ve %99.999 değeri ise 5 dakika/yıl devre dışında kalabilirliği göstermektedir.

# Sanallaştırma Yaklaşımları

---

## Donanım Seviyesi Sanallaştırma

- IBM Mainframe ve Unix/RISC sistemlerde kullanılan bölümlü yapıyı (partitioning) oluşturmak için geliştirilmiş olan sanallaştırma yaklaşımı. Donanım üreticisi tarafından sağlanan araçlar ile gerçekleştirilir.

## Yazılım Seviyesi Sanallaştırma

- X86 platformunun yaygınlaşması ile, donanım üreticisine bağlı olmayıp sadece işlemciye bağlıdır.
- Hypervisor (Bare-Metal) Mimari
  - Özel bir sanallaştırma yazılımın çıplak (bare) makine üzerine uygulanmasıdır. Tüm kontrol sanallaştırma yazılımının kontrolünde olup bu yazılım ilk elden kaynakların yönetimini sağlamaktadır. Type-1 Hypervisor olarak bilinen bu yaklaşım özellikle sunucu sanallaştırmalarında kullanılmaktadır.
- Konaklı (Hosted) Mimari
  - Daha çok son kullanıcılara hitap eden ve standart bir işletim sistemi üzerinde uygulanılabilen yaklaşımdır. Type-2 Hypervisor olarak da adlandırılmaktadır. Amacı mevcut bir işletim sisteminin üzerinde çalışan bir uygulama yardımıyla sistem kaynaklarının konak sistem tarafından da kullanılmasına imkân vererek aynı anda birden fazla sistemi çalıştırmaktır.

# Sanallaştırma Yaklaşımları

---

## Uygulama Seviyesi Sanallaştırma

- Tamamen farklı donanım platformlarında çalışabilen Yüksek Seviyeli Dil Sanal Makineleri (örneğin JVM – Java Virtual Machine) kullanılarak da uygulamalar çalıştırılabilmektedir. Bu sanal makineler yardımı ile uygulama altyapıdan ayrılmaktadır.

## Sanal Masaüstü Altyapısı (VDI-Virtual Desktop Infrastructure)

- 2006'dan itibaren başını VMware, Citrix ve Microsoft gibi üreticilerin çektiği VDI aslında sunucu taraflı kaynakların masaüstü sistemler gibi kullanılması esasına dayanır.
- Israrlı (persistent) ve Israrsız (non-persistent) olmak üzere iki alt ayrımında incelenebilir.
  - Israrlı (persistent) tipte her kullanıcı kendi masaüstünde sahip olduğu esnekliğe (program kurma, kaldırma, sistemi yönetme, dosya yaratma vb.) sahip bir sistem kopyasına (system image) sahiptir.
  - Israrsız (non-persistent) tipte kullanıcılar aynı özelliklere sahip sistem kopyalarının (system image) bulunduğu bir havuzdan hizmet almakta, işlemlerinin bitmesini ve sistemden çıkmalarını (log-out) takiben sistemler ilk hallerine geri dönmekte yani kullanıcının yaptığı hiçbir işlem kalıcı olmamaktadır.

# Bilgisayar Ağları

---

Kişisel bilgisayarların artan işlem gücüne paralel olarak bilgisayar ağları konusunda da büyük ilerlemeler yaşandı.

- Kişisel bilgisayar sistemlerinin sayısının artması ve bunlar arasındaki bilgi paylaşım ihtiyacı yerel alan ağlarının kurulmasını gerekli kıldı.

1969 yılında temelleri atılan TCP/IP tabanlı Internet ağının yaygınlaşması ile günümüzde büyüklüğüne küçüklüğüne bakılmaksızın her türlü bilgisayar sistemi birbiri ile iletişim içinde.

- Kaynak ve bilgi paylaşarak daha üretken amaçlar, katma değer sağlayan servisler ve kişisel gelişim amaçlı olarak kullanılabilir.

Bilgisayar ağları ile dağıtık olarak çalışan kişisel bilgisayar sistemlerinden oluşan (distributed) model, merkezi (centralized) yapıdaki bilgisayar sistemleri üzerine kurulu modele üstünlük sağlamıştır.

Oysa günümüzde ağa bağlı bilgisayar sistemleri kullanılarak (network computing) işlemlerin zayıf işlemciler (thin client) tarafından merkezde bulunan kuvvetli sunucular (fat server) üzerinde yapılması benimsenmektedir.

Bilgisayar ağları, İstemci/Sunucu yapısının ihtiyaçların gereğine göre kurulmasına imkân sağlamaktadır.

# İstemci Sunucu Mimarisi

---

# C/S Modeli

---

Bilgisayar topluluğu ikiye bölündü

1. Kişisel bilgisayarlarla herşey yapılabilir, merkeze gerek yok, dağıtılmış PC'lerin işlem gücü ortak bir amaca hizmet edebilir.
2. Merkezi sistemler daha da gelişti, ağlar yaygın kullanılıyor, PC'ler birer terminal gibi merkezi sistemlerin birer parçasıdır.

Her iki görüş de hem doğrulara, hem yanlışlara sahip.

C/S mimari, verimli bir kullanım şeklidir.

- Açık ve esnektir.
- Gelişen ağ altyapısına uyum sağlayabilir.
- Fiyat/performans tatminkardır.
- Mümkün olduğunca çok servis vermeye odaklıdır (PC/workstation dağıtık sistemler veya merkezi sunucular kullanılabilir)

Ağ üzerinde zayıf istemciler, merkezdeki kuvvetli sunucularda işlemleri yaptırır.

# C/S Sistemlerin Özellikleri

---

1. **Servis** (Service): İstemci/Sunucu sistemlerin amacı bir servis vermektir. Bu amaçla tasarlanmış iki parçadan oluşmuştur. Sunucu, servisi veren kısım, istemci ise bu servisten yararlanmak isteyen kısımdır.
2. **Paylaşım** (Shared Resource): Sunucu, aynı zamanda pek çok istemciye hizmet verebilmekte, kendi kontrolü altındaki kaynağı istemcilerin istekleri doğrultusunda sunmaktadır. Bu sayede sunucu, istemcilerin kendi kontrolündeki kaynaklara kontrollü erişimini sağlar.
3. **Asimetrik Protokol** (Asymmetric Protocol): Sunucu ile istemciler arasında tek-çoklu ilişki vardır. Sunucu, istemcilere bir servis vermek için beklerken servis almak isteyen istemciler de sunucunun verdiği servisten yararlanmak için istekte bulunacaklardır.
4. **Yerden Bağımsız** (Transparency of Location): İstemci ve sunucu parçalarının hangi bilgisayar üzerinde çalıştığının pek bir önemi yoktur. Bunların ikisi de gerekiyor ise aynı bilgisayarda çalışabileceği gibi farklı bilgisayarlar üzerinde de çalışabilirler. Hatta sunucunun yeri çeşitli yönlendirme yöntemleri ile (re-direction) ile gizlenebilir.



# C/S Sistemlerin Özellikleri

---

5. **Farklılık ve Uyum** (Mix & Match): İstemci/Sunucu sistemler; kullanılan donanım platformlarından, işletim sistemlerinden bağımsız olarak çalışmak üzere tasarlanırlar. Buna göre sunucu parçası RISC mimari modelinde tasarlanmış işlemciye sahip Unix işletim sistemli bir bilgisayar sistemi üzerinde çalışırken istemci parçası CISC mimari modelinde tasarlanmış işlemciye sahip Windows işletim sisteminden çalışabilir. Önemli olan bunların bir araya geldiğinde uyum içinde amaçlanan görevi tamamlamasıdır.
6. **Mesaj Tabanlı Haberleşme** (Message Based Exchange): İstemci ve Sunucu birbirleriyle seyrek bağlantılı (loosely coupled) olarak çalışmaktadır. Bu nedenle aralarından istek/cevap (request/reply) tipinde bir mesaj alışverişi yapılarak görev gerçekleştirilmektedir.
7. **Servisin Veriliş Şekli** (Encapsulation of Service): Sunucu, istemciden aldığı istek üzerine görevi yerine getirmek için hareket eder. Gelen istek için gerekli cevabın üretirken nelerin, nasıl yapılacağı sunucun sorumluluğunda olup detaylar istemciyi hiçbir şekilde ilgilendirmez.

# C/S Sistemlerin Özellikleri

---

8. **Ölçeklenebilirlik** (Scaleability): Sunucu sistemler artan yüke bağlı olarak iki ayrı şekilde ölçeklenebilir.
  - a. **Dikey** (Vertically): Sunucunun çalıştığı donanımın daha hızlı bir modele çıkartılması (**scale up**): Duruma göre aynı donanım üzerinde bellek artımı, işlemcinin daha hızlısı ile değiştirilmesi, ek işlemci kullanılması, çevre birimlerin kapasite ve hızlarının arttırılması ile yapılabileceği gibi, mevcut donanımın değiştirilerek daha üst performans sağlayacak bir modele geçilmesi olarak da uygulanabilir. Günümüzde aynı donanım üzerinde **sanallaştırma** (virtualization) teknikleri kullanılarak elde olan donanımın daha etkin kullanılmasına yönelik çalışmalarda (her ne kadar yeni bir sunucu ekleniyor gibi görünse de mevcut kaynaklar kullanıldığı düşünülerek) dikey ölçekleme olarak yorumlanmaktadır.
  - b. **Yatay** (Horizontally): Sunucu yükünü mümkün olduğunca başka sunucular üzerine aktarılması (**scale out**): Bu tür bir uygulamada sunucu kümeleri (cluster, server farm) oluşturulması ve yük paylaştırılma yazılımları veya donanımlarından yararlanılması gereklidir. Uygulamanın özelliklerine bağlı olarak tek arıza noktası (single point of failure), yük dengeleme (load balancing), dağıtım (decentralization), eş zamanlama (synchronization) ve yineleme (replication) konularına dikkat edilmesi gereklidir.
9. **Bütünlük** (Integrity): Sunucu tarafındaki kod ve veri üzerinde merkezi bir yönetim uygulanmaktadır. Bu sayede sistemin bakım maliyeti azaltılır. Bilginin yedeklenmesini kolaylaştır. İstemci/Sunucu uygulamaların amaca uygun tasarlanmış olması sistem ve/veya uygulama yazılımlarında herhangi biri açık olmaması durumunda bilgi güvenliği de sağlanmış olur.

# C/S Uygulamaları

---

- 1. Dosya Sunucusu (File Server):** İstemci, ihtiyaç duyduğu dosyanın ismini bir istek (request) olarak sunucuya iletir. Sunucu ise paylaşım sunduğu kaynaklarından (disk) bu bilgiyi istemciye iletir. Sunucunun, istemci adına sunulan dosya üzerinde herhangi bir işlem (hesaplama) yapması söz konusu değildir. İş yükü istemci taraflıdır.
- 2. Yazıcı Sunucusu (Print Server):** Genellikle ağ üzerinde yer alan ve üzerinden bir veya daha fazla yazıcıya erişim imkânı veren cihazları ifade eder. LPD/LPR, TCP Port 9100 ve RAW protokollerini kullanarak dosyanın seçilen yazıcıdan basılmasını sağlar.
- 3. Veri tabanı Sunucusu (Database Server):** Sunucu tarafı veri tabanından almak istediği sonucu elde etmesini sağlayan SQL cümlesini bir istek olarak iletir. Sunucu tarafında üzerinde işlem yapılacak veri dışında SQL cümlecini anlayıp sonucu üretmeyi sağlayacak olan program da bulunmaktadır. Sunucu bu işlemi yerine getirmek için kendi kaynaklarını kullanır (işlemci, bellek vs.)

# C/S Uygulamaları

---

- 4. Hareket Sunucusu** (Transaction Server): İstemcinin, sunucu tarafında bulunan uzak yordamları (remote procedure) kullanarak görevini yerine getirilmesi prensibine dayanmaktadır. Bu tip uygulamalarda istemci tarafında da sunucu tarafında da gerekli kodlamaların yapılması gereklidir. İstemci yolladığı istek ile bir yordam çağırarak bir grup SQL cümlesini çalıştırabilir. Buna rağmen dönecek sonuç, SQL cümleleri grubunun tamamının başarı olup olmadığını belirleyen tek bir değerdir. Bu tür uygulamalar OLTP (On-Line Transaction Processing) olarak adlandırılmaktadır. Bir banka uygulamasında müşterinin (mudi) vadesiz döviz hesabından 100 € bozdurarak oluşan TL değerinin 100TL'sini bir başka müşterinin vadesiz Türk Lirası hesabına aktarması işlemi incelediğimizde;
- 100 € müşterinin vadesiz döviz hesabından çekilir, müşterinin hesap bakiyesi güncellenir.
  - 100 € bankanın vadesiz döviz hesabına aktarılır, hesap bakiyesi güncellenir.
  - 1 € nun TL karşılığı tespit edilir.
  - Bankanın vadesiz döviz hesabındaki 100 € çekilip (bakiye güncellenir) TL dönüşümü yapılarak müşterinin vadesiz TL hesabına karşılığı aktararak bakiye güncellenir.
  - Banka döviz işlemleri için uygulayacağı komisyon/vergi vb. kesintileri hesaplayarak bunları kullanıcının vadesiz TL hesabından çekerek bankanın vadesiz TL hesabına aktarırken her iki hesabın bakiyelerini günceller
  - Müşterinin vadesiz TL hesabından 100 TL çekilip bankanın kasasına aktarılıp (para aktarımı için uygulanacak komisyon var ise bu değerde kullanıcının vadesiz TL hesabından çekilecektir) ve bakiye güncellenir.
  - Bankanın vadesiz TL hesabından 100TL alacaklı olan müşterinin vadesiz TL hesabına aktarılırken her iki hesap bakiyesi güncellenir.

Bu adımların hepsinin hatasız ve bölünmeden yapılması (atomic) durumunda işlem başarılı olmuş sayılacaktır. Aksi halde herhangi bir maddi kayıp olmaması amacıyla işlemlerin işlem başlamadan önceki aşamaya döndürülmesi (roll back) gereklidir.

# C/S Uygulamaları

---

5. **Grup Sunucusu** (Groupware Server): Metin, resim, e-mail, haber tahtası, iş akışı gibi kısmi organize edilmiş bilginin kullanımı ve yönetimi amacıyla kullanılırlar. Bu tip uygulamalar tanımlı gruplar içinde yer alan kullanıcıların birbirlerini yaptıklarından haberdar etmelerini ve birbirleri ile bağlantı içinde olmalarını sağlamak üzere kullanılmaktadır. IBM Lotus, Microsoft Exchange Server ve Microsoft SharePoint bu tür uygulamaların en iyi örneklerini teşkil etmektedir.
  - Bu uygulamalar kendi içlerinde farklı modüllere sahip olmak ile birlikte genelde Wiki, Web Publishing, Calendar, Project Management, WorkFlow, Document Management, List Management, XML, Blog, Survey, Time Tracking, E-mail, Fax, Instant Messaging, Video Conferencing, Application Sharing, Electronic Meeting gibi farklı servisleri verebilmektedir.
6. **Nesne Uygulama Sunucusu** (Object Applications Server): Nesneye yönelik tasarımın bir sonucu olarak İstemci/Sunucu uygulamalar birbirleri ile iletişim kurabilen nesneler olarak tasarlanmaktadır. İstemci nesneleri, ORB (Object Request Broker) adı verilen bir yapı yardımıyla sunucu nesneleri ile iletişim kurmaktadır. İstemci uzak nesne (remote object) üzerinde tanımlı bir metodu kullanmak istediğinde ORB devreye girerek bu nesnenin bir örneğini (instance) oluşturan sunucu nesnesini bulur, istenilen metodu çağırır ve sonucunu istemci nesnesine döndürür.
7. **Web Uygulama Sunucusu** (Web Application Server): Web uygulamaları en yaygın İstemci/Sunucu uygulamalarıdır. Basit, kodu farklı yazılım ve donanım platformlarında kullanılabilir (taşınabilir- portable) ve evrensel bir istemci yazılımın, çalışan bir sunucu (superfat server) üzerinden servis alması prensibine dayanmaktadır. Web Server, istemci tarafından istenilen bilgileri iletmek üzere hazırlanmıştır. Aradaki iletişimi, basit bir komut seti kullanan ve aktarılan parametreleri dizgi (string) olarak karşı tarafa yollamayı sağlayan HTTP (Hyper Text Transfer Protocol) sağlamaktadır.

# Servis yerine Yük Temelli Sınıflandırma

Groupware, Transaction ve Web Server uygulamalarda işlem yoğunluğu sunucudadır.

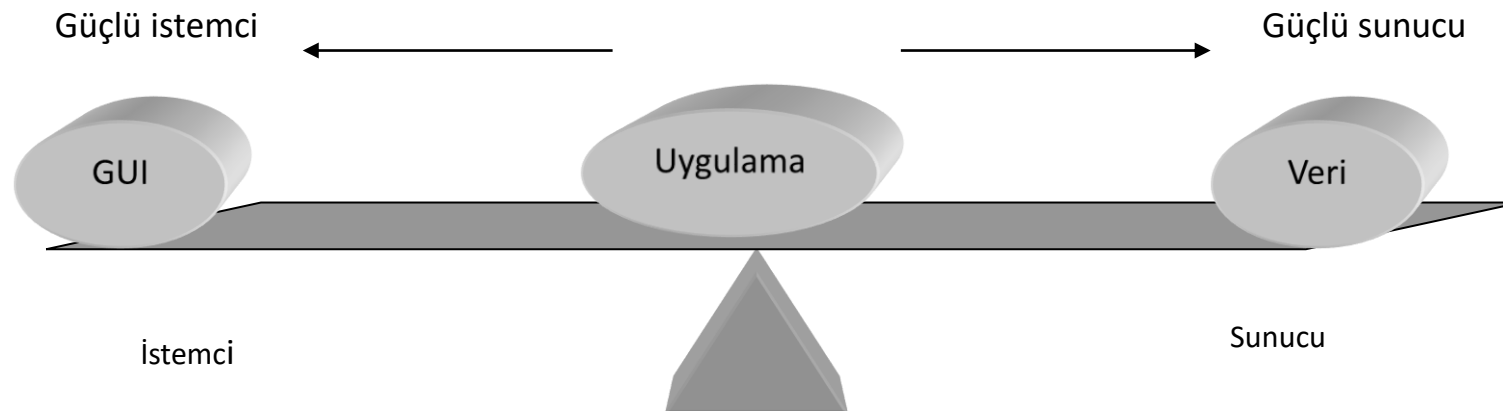
- Yönetim daha kolay.

Database ve File Server uygulamalarında işlem yükü istemcidedir.

- Eski, alışlagelmiş uygulamalar. İstemci, veri organizasyonunu hakkında bilgili olmalıdır.

Zayıf istemci: Mainframe/terminalin devamı. Input/output harici bir işlem istemcide yapılmaz.

Sıfır istemci (zero client): System on a Chip



# İstemci Çeşitlerinin Kıyaslanması

Özellik	PC (Thick Client)	Zayıf İstemci (Thin Client)	Sıfır istemci (Zero Client)
Maliyet	~ 500 USD	~ 250 USD	~ 300 USD
İşlemci	İhtiyaca göre birkaç GHz hızlarında piyasada bulunan x86 işlemcilerden	400-800 MHz hızlarında piyasada bulunan x86 işlemcilerden	Grafik işlemleri hızlı yapan özel tasarım TERA2321, TERA2140 işlemci
Bellek	İhtiyaca göre birkaç GB	32-128 MB	512 MB
Disk (Depolama)	İhtiyaca göre GB-TB	İşletim sistemi ve kullanıcı ara yüzü için 32-64MB Compaq Flash	Yok
Ekran	İhtiyaca/donanıma göre 1-2 tane yüksek çözünürlüklü	1 tane orta çözünürlüklü	2 veya 4 tane yüksek çözünürlüklü
İşletim Sistemi	İhtiyaca bağlı olarak Windows, Linux vb.	Windows CE veya mini Linux Kernel (fiyata dâhil)	İşlemci üzerinde (fiyata dâhil)
Ek yazılım	İhtiyaca bağlı uygulamalar kurulmalıdır.	Uygulamalar merkezi sunucu üzerindedir.	Uygulamalar merkezi sunucu üzerindedir.
Yönetim ve kontrol	Donanım/Yazılım kurulumu, yönetimi ve güncellenmesi kullanıcı sorumluluğundadır. Destek hizmetinin uzaktan verilmesi durumunda ağ üzerinde trafik oluşmaktadır.	Uygulamalar sunucu taraflı çalışır. Cihaz üzerinde farklı protokoller ile sunucuya bağlanmayı sağlayan yazılımın koşullandırılması gerekir.	PCoIP ile sunucu bağlantısı sağlanır. Ek olarak yapılması gereken bir faaliyet yok.
Veri yedekleme ve güvenlik	Veri yedeklemesi kullanıcı sorumluluğundadır. Virüs vb. tehditlere açıktır. Kullanıcı sistem ayarlarını değiştirerek çalışmayı etkileyebilir. Disk, CD/DVD gibi dönerek çalışan parçalar ısınma ve sese neden olur	Veri merkezde saklanır yedeklenir. Kullanıcının sistem ayarlarını bozma olasılığı düşüktür. Donanım olarak dönen birimler olmadığı için daha az enerji tüketir ve daha uzun ömürlüdür.	Veri merkezde saklanıp yedeklenir. Yerel olarak çalışan uygulama yoktur. Donanım olarak dönen birimler olmadığı ve amaca uygun tasarlanmış özel işlemci sayesinde daha az enerji tüketir ve daha uzun ömürlüdür.
Tüketim	150-200 W	20-30 W	6-12 W
Ömür	2-3 yıl	5-7 yıl	> 7 yıl

# 2 ve 3 Katmanlı C/S Uygulamalar

Fark: Uygulamanın ne tür işlevsel parçalara ayrılacağı ve bunların hangilerinin istemci tarafında, hangilerinin sunucu tarafında kalacağını belirlemesidir.

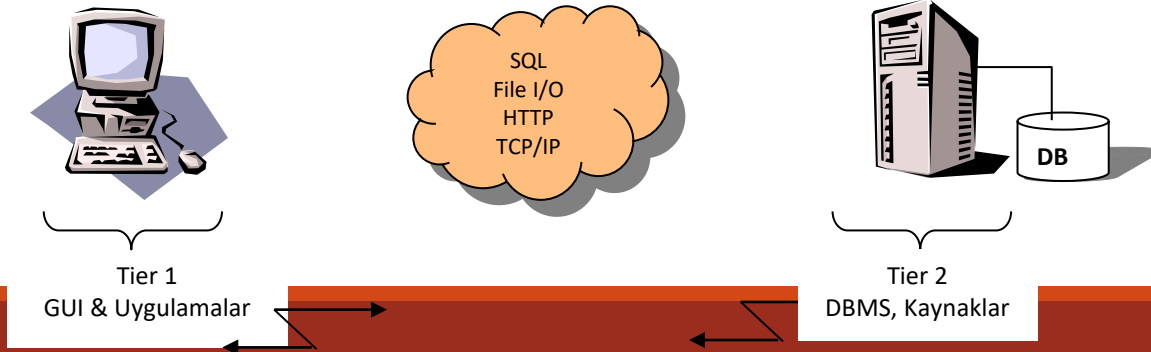
İşlevsel parçalar: kullanıcı ara yüzü (user interface), iş mantığı (business logic) ve paylaşılan veri.

Uygulama istemci üzerinde çalışan kullanıcı ara yüzüne veya sunucu üzerindeki veri tabanına, bazı durumlarda da her ikisine birden gömülü durumdadır.

- Dosya sunucular ve veri tabanı sunucuları (stored procedure kullanan)

Güçlü istemci türündedir. Yerel alan ağı sınırları dışından kullanılması oldukça büyük yönetim sorunlarının çıkmasına neden olabilir.

- Değişikliklerin devreye girebilmesi için tüm istemcilerde bu işlemlerin tekrar tekrar yapılması gerekli.





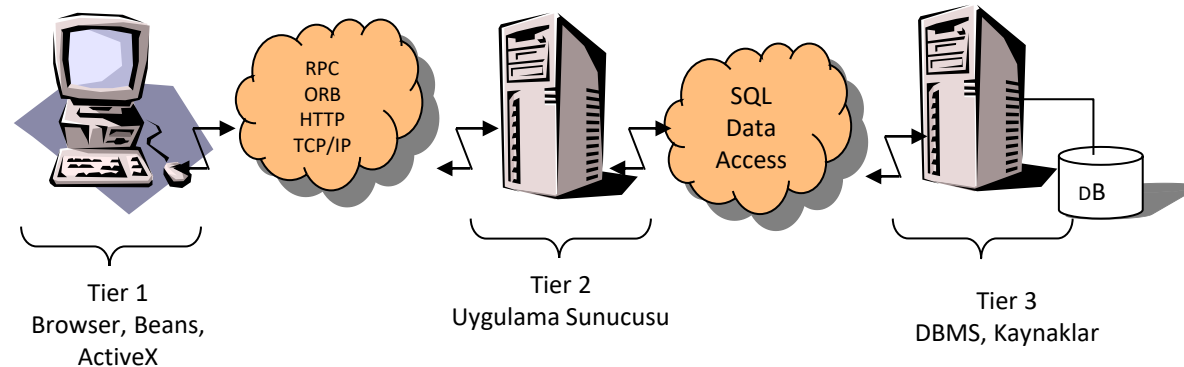
# 2 ve 3 Katmanlı C/S Uygulamalar

3 katmanlı modelde istemci (1.kat), uygulama sunucusu (2.kat) ve nihayet veri tabanı sunucusu (3.kat) olarak görülmektedir.

Orta katman ölçeklenebilir (scaleable), sağlam (robust) ve esnektir (flexible).

En basit intranet uygulamalardan en geniş ölçekli internet uygulamalarına kadar ideal çözüm olmuştur.

işin büyük kısmının ortada bulunan uygulama sunucusunda çalışması sayesinde gerek bakım gerekse yönetimi de kolaydır.



# 3 Katmanlı Model ve MVC Mimarisi

---

Sunum (presentation logic) ile arka planda yürütülen iş mantığı (business logic) kısımları birbirinden ayrıdır. Geliştirilme, test ve bakım süreçleri bağımsız yapılabilir.

**Model :** Model, iş mantığının davranışının (behaviour) yönetilmesi ve veri nesnelerinin (data objects) sunulması için kullanılmaktadır. Model kullanıcının isteğine karşı düşen cevabı sunmak için yapılacak işlerin tariflendiği kısım olarak karşımıza çıkar. Bu nedenle Model, veri ve verinin uygulama içinde nasıl işleneceğini tarifleyen iş mantığı olarak düşünülmektedir.

**View:** Model gereğince kullanıcı ile etkileşimi sağlayacak biçimde sunulan görüntü View olarak adlandırılır. Veri nesnesinin o andaki durumunu kullanıcıya göstermek amacıyla kullanılır.

**Controller:** Kullanıcı etkileşimi ile alınan giriş bilgileri alarak, Model nesnesi üzerinden gerekli cevabın oluşturulmasından sorumludur. Kullanıcı arayüzünden gelen giriş bilgileri doğrultusunda ne şekilde hareket edeceği ve cevabın nasıl üretileceğinden sorumludur.

Web uygulamalarında View bir uygulama tarafından oluşturulan HTML, Controller GET ve POST yapısındaki kullanıcı etkileşimini alan ve bunun ile ne yapılması gerektiğine karar veren kısım, Model ise veri ile veri kullanılarak gerçekleştirilecek olan iş mantığıdır.

# MVC Mimarisinin Avantajları

---

**Güvenilirlik (Reliability):** Sunum ve iş mantığı için gerçekleştirilen işlemlerin birbirinden ayrılması ile uygulamanın görünen yüzü olan sunumun değiştirilmesi model ve controller kodunun derlenmesini gerektirmez.

**Yeniden kullanılabilirlik ve uyumluluk (High reuse and adaptability):** MVC farklı tipteki görüntüler kullanmasına rağmen aynı sunucu taraflı kodu kullanabilir. Bu sayede gerçekleştirilen iş mantığı ön planda kullanılan cihaza bağlı kalmadan (web browser, mobil gibi) sadece kullandığı sunum ön yüzü değiştirilerek değişik platformlarda servis verebilir hale gelir.

**Kısa geliştirme süresi ve düşük hayat/maliyet (life/cost) değeri:** MVC modeli sayesinde kullanıcı arayüzünün tasarlanması sırasındaki sürenin kısılması ve bu işi yapacak kişilerin çok bilgili programcılar olmasının gerekmemesi hem hız hem maliyet kriterleri için önemlidir.

**Bakım kolaylığı (maintainability):** Sunum ve iş mantığı arasındaki ayrılık özellikle web uygulamalarının bakımını çok büyük oranda kolaylaştırır.

# N Katmanlı Mimariler

---

Orta katman tek parça (monolithic) olarak tasarlanmaz, farklı istemci isteklerine cevap verecek şekilde düzenlenmiş parçalar halindedir. Bu parçalardan herhangi biri kendi üzerine düşen küçük bir işi yerine getirir. İstemcilerin isteğini yerine getirmek üzere bu parçalardan bir grubu kullanılır. Bunun için bir parçanın bir diğer parçayı da çağırması gerekebilir.

Büyük uygulamaların küçük adımlar ile gerçekleştirilmesini sağlar.

Uygulama parçalarının (components) tekrar kullanabilmesi sayesinde sağlar.

Kullanıcı, isteğine üretilen cevabın arka planda nasıl (hangi veri tabanına erişilerek, ne tür işlemlerden geçtikten ve ne tür bir mantıksal süzgeçten geçtikten sonra) hazırlandığını bilmez.

Özel uygulamalar var olan parçalar kullanılarak gerçekleştirilebilir. Ancak yeni uygulamalar için var olan parçaların bir araya getirmesini sağlayacak mantıksal yapı kurulmalıdır.

Parçalar zaman içinde eskimezler ancak yapılabilecek iyileştirmeler sayesinde daha kusursuz hale gelirler. Bu iyileştirmeler o parçaların kullanıldığı diğer uygulamaların da daha etkin hale gelmelerine yardımcı olur.

# Peer-to-Peer (Eşdüzey) Uygulamalar

---

Haberleşme içinde olan birimlerin net tanımlanmış, değişmez işlemci ve sunucu gibi ayrı fonksiyonel özellikleri yoktur.

Her üye duruma göre sunucu, duruma göre istemci rolünü oynayabilir. Gerçek İstemci/Sunucu uygulamalarda gerek istemcinin gerek ise sunucunun rolleri belli ve değişmezdir.

Eşdüzey uygulamaları çalıştıran sistemler kendi sistem kaynaklarını (disk, işlemci vb) bir kısmını doğrudan ağ üzerinden kendisine erişenlerin hizmetine sunar.

Özellikle dosya alış-verişi amaçlı olarak kullanılmaya başlanarak yaygınlaşmıştır. Peer-to-peer uygulamalar, paylaşılan dosyanın hangi sistemler üzerinde yer aldığı bilgisini tutan bir takım rehber servisler vasıtası ile birbirlerini bulup daha sonra kendi aralarında veri alışverişini sağlarlar.

## 2-3 Katman Karşılaştırması

Özellik	2 Katman	3 Katman
Sistem Yönetimi	Zor	Kolay
Güvenlik	Düşük	Yüksek
Veri Sunuş Biçimi (encapsulation)	Sarmalanmamış	Sarmalanmış
Başarım	Düşük	Yüksek
Ölçeklenme	Zayıf	Mükemmel
Tekrar Kullanılabilirlik	Az	Yüksek
Geliştirme Kolaylığı	Yüksek, hızlı	İyileşiyor
Eski Uygulamalara Uyumluluk	Yok	Mümkün olabilir
İnternet Desteği	Zayıf	Yüksek
Ayrı Cins Veritabanı Desteği	Yok	Mümkün
İletişim Seçenekleri	Sadece senkron RPC	connectionless da olabilir
Donanım Esnekliği	Sınırlı	Yüksek

# C/S Yapıtaşlarının Özellikleri

---

## C: İstemci :

- Grafik ara yüzü olmayanlar (metin tabanlı)
- Grafik arayüzü olanlar (basit)
- Nesne tabanlı grafik arayüzü olanlar (sürükle bırak, Java desteği vs)

## S: Sunucu:

- Ortaklaşa kullanılan kaynakların bütünlüğünü (integrity) sağlayarak aynı anda (concurrent) pek çok isteği karşılamak
- İşlem öncelikli istemcilere servis vermek
- Gerektiğinde arka plan işleri koşturmak
- Sağlam donanım altyapısı, ölçeklenebilirlik

## /: Orta Katman:

- Taşıma yığını (Transport Stack: TCP/IP, IPX/SPX, SNA, NetBIOS gibi)
- Ağ işletim sistemi (Network Operating System: NOS)
- Dağıtık sistem yönetimi (Distributed System Management (DSM): SNMP, CMIP gibi )
- Ara parçaya has servisler (Service Specific to Middleware: ODBC, Mail, ORB, http gibi)

# Gelecek Ders

---

Web programlama HTML/ CSS

Proje konu tanıtımı ve atama