

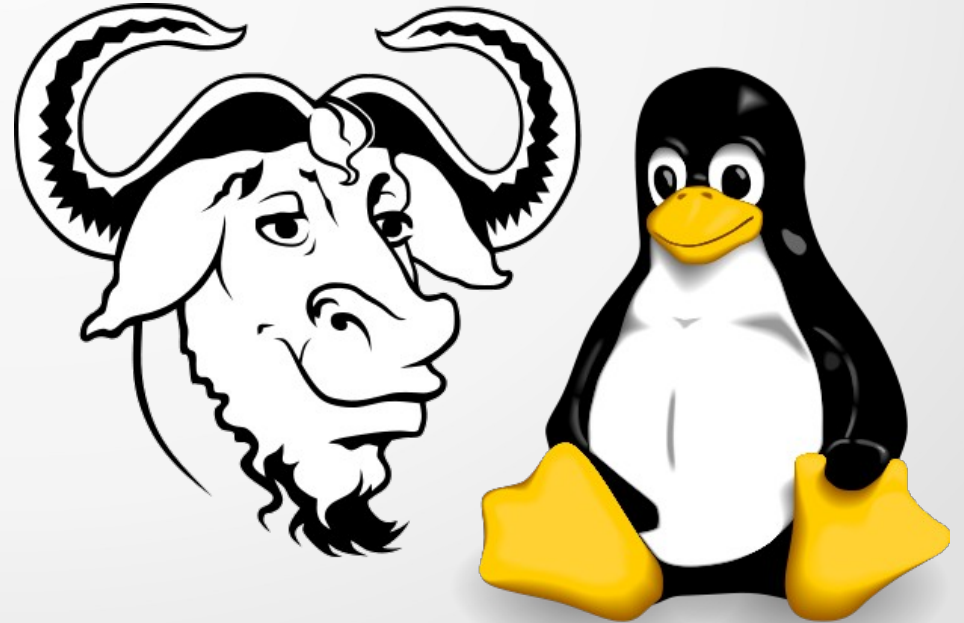


# PARDUS

## Özgür Yazılım Felsefesi ve Temel Kavramlar

Mustafa Akgül Kış Kampı 2019

**Eğitmenler:** Duygu Ölmez  
Emre Kondul



## Özgür yazılım vs. Açık Kaynak

- Açık kaynak kodlu bir yazılım, özgür olmayabilir.
- Özgür bir yazılım, açık kaynak kodludur.
- **Açık kaynak**
  - Teknik avantajlara odaklanır. Yazılımı açık kaynak olarak geliştirmek teknik olarak daha başarılı bir model olduğu için tercih edilir.
- **Özgür yazılım**
  - Teknik avantajların yanı sıra işin toplumsal boyutunu ve "özgürlük" kavramını da ele alır.

## Yazılımın Özgürlüğü Kavramı?

Yazılım *kullanıcısının*, yazılım üzerinde dağıtma, değiştirme, inceleme gibi haklara sahip olduğu yazılımdır.

**Özgürlük 0** - İstenen her amaca uygun olarak uygulamayı **çalıştırabilmek**,

**Özgürlük 1** - Programın çalışmasını inceleyebilmek ve kendi ihtiyaçlarına uygun biçimde **değiştirebilmek**,

**Özgürlük 2** - Programın kopyalarını **dağıtabilmek**,

**Özgürlük 3** - Program geliştirebilmek ve bu geliştirmeleri **halka açarak** tüm camianın faydalanabilmesini sağlamak.

Bir yazılım, eğer yazılımın kullanıcıları bu özgürlüklere sahipse özgür yazılımdır. Yani, kopyalarını değiştirerek ya da değiştirmeden, **ücretli** ya da **ücretsiz**, herhangi birine dağıtmaya serbest olmalısınız. Bu anlamda özgür olmak demek, tüm bunları yapmak için izin istemek zorunda olmamanız demektir.

## Özgür Yazılımın Önemi

**Güvenilirlik :** Açık kaynak kodlu yazılımların belki de en büyük artısı olan güvenilir oluşları, kimi çevreler tarafından doğru olmayan bir mantıkla bir dezavantaj gibi gösterilmeye çalışılmakta ve topluma 'Açık kaynak kodlu bir yazılımın tam olarak nasıl çalıştığı kaynak koduna bakılarak bilinebildiği için yazılımın barındırdığı güvenlik problemleri saldırganlar tarafından kolaylıkla tespit edilip kullanılabilir' gibi bir düşünce yerleştirilmeye çalışılmaktadır. Fakat açık kaynak kodlu bir yazılım, doğası gereği geliştirilmesi sürecinden kullanımı zamanına değin onbinlerce göz tarafından denetlenmektedir. Var olması muhtemel bir güvenlik problemi ya da kötü niyetli bir programcı tarafından bilinçli olarak yerleştirilebilecek bir kod parçası yazılım sizin elinize ulaşmadan tespit edilmekte ve hızla düzeltilmekte ya da size ulaşması engellenmektedir. Ayrıca kimse, binlerce insanın kollektif bir şekilde geliştirdiği bir yazılımın güvenilirmez olduğunu idda ederken, sınırlı sayıda kişinin geliştirdiği ve kaynak kodunu kapattığı bir yazılımın güvenilirliğini kimin ve nasıl garanti edeceği sorusuna yanıt veremez.

**Sağlamlık :** Yine açık kaynak kodlu yazılımların doğal geliştirme süreci içerisinde çok sayıda insan tarafından denenmeleri ve kaynak kodlarının gözden geçirilmesi sonucunda ortaya yüksek kalitede, stabil ve kuvvetli yazılımlar çıkmaktadır.

**Esneklik :** Kaynak kodu açık bir yazılım hızla ve kolaylıkla yeni bir sistem üzerinde çalışacak şekilde yeniden yapılandırılabilmekte, bir kısmı çıkarılarak kapsamı daraltılabilmekte ya da eklenen yeni fonksiyonlarla kapsamı genişletilebilmektedir.

**Uygulama Desteği :** Çok geniş bir yelpazede ve çeşitlilikteki bir çok açık kaynak kodlu özgür yazılım her gün duyurulmaktadır. Herhangi bir konudaki ihtiyaç açık kaynak kodlu yazılımlarla hızlı ve kaliteli bir şekilde çözüme kavuşturulabilmektedir.

## Yazılım Lisansları

- Özgür yazılımların tek çeşit lisansı yok
- Farklı kullanımlara özelleşmiş çeşitli lisanslar var
- Çoğu lisans aslında birbirinine çok benzer

### **GPL: (Genel Kamu Lisansı)**

- Özgür yazılımların %66'sı tarafından tercih ediliyor.
- Yazılımın özgür olmasının yanı sıra, özgürlüğünü sürdürmesini de garanti altına alır.

### **LGPL: (Kısıtlı Genel Kamu Lisansı)**

- GPL'den farklı olarak, yazılımın özgür olmayan yazılımlar tarafından da kullanılabilmesine olanak tanıyor.

### **Apache Lisansı**

- GPL'e ek olarak, patentlere yönelik belirli kısıtlamalar içeriyor.

### **Ve daha bir çok Lisans bulunmakta**

<https://www.gnu.org/licenses/license-list.html>

<http://ozgurlisanslar.org.tr/>

# Programlama Dilleri ve Araçları

## Programlama Dilleri



## Yazılım Geliştirme Araçları



## Başlıca Ö/AKK yazılım projelerine örnek olarak

- GNU/Linux işletim sistemi,
- Mozilla Firefox ağ tarayıcısı,
- Apache ağ sunucusu,
- Samba dosya paylaşım sunucusu,
- MySQL ve PostgreSQL veritabanı sunucuları,
- GNOME ve KDE masaüstü ortamları,
- GIMP grafik düzenleme yazılımı,
- PHP, Perl ve Python programlama dilleri,
- BIND alan adı sunucusu,
- Sendmail e-posta sunucusu (zimbra),
- Mozilla Thunderbird e-posta istemcisi,
- LibreOffice

Mozilla Firefox ve Thunderbird, Python, LibreOffice gibi çok sayıda Ö/AKK yazılımın aynı zamanda Microsoft Windows gibi özel mülk işletim sistemlerinde de çalıştırılabilen sürümleri bulunmaktadır. Bu durum Ö/AKK yazılımın kullanıcılar arasında bir ayrımcılık yapmadığını ve özel mülk yazılım kullanıcılarını dışlamadığını açık bir biçimde göstermektedir.

Ö/AKK- Özgür ve Açık Kaynak Kod

# Programlama Dilleri ve Araçları

## Java

Java programlama dili 1995 yılında, geliştirilen bir uygulamanın üzerinde uğraşılmadan farklı platformlarda çalışabilmesini ve yazılımcıların hata yapma ihtimalini azaltma amaçlarıyla geliştirilmiş, günümüzde ise en yaygın kullanılan programlama dillerinin başında gelmektedir. Mobil cihazlardan masaüstüne, sunuculardan özel geliştirilmiş cihazlara kadar bir çok farklı platformda ve işletim sisteminde çalışabilen java uygulamaları, günümüzde özellikle "kurumsal uygulamalar" için en çok tercih edilen programlama dilidir.

## Ruby

İlk ortaya 1995 yılı civarında çıkmış olmasına rağmen 2000'ler sonlarına doğru popüler olan bir programla dilidir. Dinamik bir dil olması nedeniyle özellikle Web uygulamaları geliştirilmesi sırasında tercih edilen, oturmuş ve yaygın kullanılan bir dildir.

## PHP

Web uygulamaları ve sayfaları için ilk kullanılan programlama dillerinden biridir. Hedefi web sayfalarının daha kolay yazılmasını sağlamak olan PHP, günümüzde özellikle web sayfaları ve uygulamaları için en çok tercih edilen programlama dili olma özelliğini korumaktadır.

## JavaScript

Tarayıcılar içinde çalışmak üzere geliştirilmiş bir betik dili olan JavaScript'in ( asıl adı EcmaScript ) zaman içinde gelişmesi ile günümüzde sunucu tarafında da kullanılmak üzere evrimleşmiştir. Özellikle Node.JS ve veri tabanlarının desteği nedeniyle, hızlı web servisleri geliştirmek için tercih edilen bir dil olmuştur.

## Python

Genel amaçlı dinamik bir dil olan Python, güçlü matematik kütüphaneleri ile akademik çalışmalarda en çok tercih edilen dillerden biri olmasının yanı sıra, işletim sistemi bileşenleri ve web uygulamaları için de yaygın olarak kullanılmaktadır.



# Programlama Dilleri ve Araçları

## Git

Günümüzde yazılan bir uygulamanın sürüm takibi yazılım geliştirme sürecinin en önemli parçasını oluşturmaktadır. Bu noktada dağıtık bir sürüm takip sistemi olan git ve web tabanlı sunucu ortamı sağlayan GitLab yazılım geliştirme sürecinin en önemli aracı olmaktadır.

## RedMine

Yazılım geliştirme sürecini takip etmek için bir çok araç ve yöntem bulunmaktadır. Bunların içinde iş takibi, ekip çalışması için büyük önem taşımaktadır. RedMine web tabanlı bir araç olarak bu konuda rüştünü ispatlamış yaygın kullanılan bir yazılımdır.

## Jenkins

Her ne kadar derleme kavramı programlama dillerine ait olarak görünse de uygulama kodlarının sadece makine diline çevrilmesi değil, uygulamanın yüklenmeye hazır paket haline getirilmesini de kapsar. Bu işlemin parçaları içinde testlerin çalıştırılması, statik kod analizi yapılması, hatta test ortamlarına uygulamanın yüklenmesine kadar bir çok işlevi yerine getirebilen Jenkins günümüzde en yaygın kullanılan düzenli derleme aracı konumundadır.

## SonarQube

Kod kalitesi ve uygulamanın sürdürülebilir olması açısından, yazılım ekibinin belirlenmiş yazım kurallarına uyması gerekmektedir. SonarCube ile bu yazım kuralları otomatik denetlenmekte ve uygunsuz durumlar önerilerle birlikte raporlanmaktadır.

## Özgür Yazılımın Önemi

- Geliştirme ve kullanabilme özgürlüğünün geliştirilmesi
- Tekerleğin tekrar tekrar icat edilmesinin önüne geçilmesi ve birikimli ilerleme mantığına dayanması
- Birlikte üretme kültürünü aşılması
- İnsanlığın bilgisinin korunması



VS.



## GNU (GNU is Not Unix) nedir?



AT&T Bell Laboratuvarları multics projesini bıraktıktan sonra AT&T Bell Laboratuvarları çalışanlarından Dennis M. Ritchie ve Ken Thompson yeni bir işletim sistemi çalışmalarına başladı. Bu işletim sistemine “Unics” ismini koydular. Daha sonra isim “UNIX” olmuştur.

UNIX işletim sistemi üniversitelerde duyulmaya başlandı. Böylece üniversitelerin katkıları oldu. 1980’li yılların başlarında AT&T şirketi para kazanma amacıyla UNIX işletim sistemini pazarlamaya başladı. UNIX’in gelişmesine yardımcı olanlar bu durumdan hoşlanmadı. **Richard Stallman** tarafından UNIX’ten bağımsız olan GNU projesi başlatılmıştır. GNU araçları UNIX kodlarını içermemektedir.

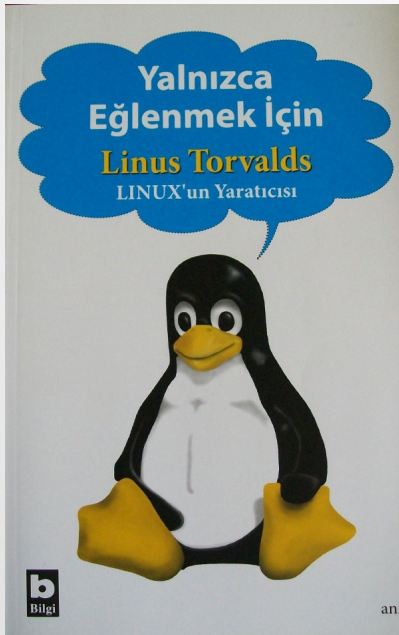
1984 yılında Stallman ve arkadaşları tarafından FSF projesi başlatılmıştır. FSF’nin açılımı “Free Software Foundation” yani “Özgür Yazılım Derneği”dir. FSF’nin amacı özgürce dağıtımı yapılabilecek ve kodları üzerinde değişiklik yapılabilecek GNU projesine destek vermektir.

# Linux Nedir?

Helsinki Üniversitesi'nde bir öğrenci:

## **Linus Torvalds** **Revolution OS filmi**

Linus'un 25 Ağustos 1991 tarihinde minix mail listesine gönderdiği maili: "Minix kullanan dışarıdaki herkese merhaba. 386(486) AT kullanan insanlar için (serbest) işletim sistemi yapıyorum. (sadece hobi, gnu gibi büyük ve profesyonel olmayacak)."



Peki Linux tam olarak ne?

Linux, aslında GNU/Linux işletim sistemlerinin sadece **çekirdeğine (kernel)** verilen isim.

Linux çekirdeğinin üzerine bir çok program eklenirse (ki bu programların bir kısmı GNU projesi araçlarını içermekte) bir Linux Dağıtımı oluşturulmaktadır.

Örneğin bir dizindeki dosyaların listelenmesi komutu olan “ls” bile çekirdeğin bir parçası değildir.





## Neden GNU/Linux?

Her birey ve kurum için çok farklı nedenler vardır fakat;

- Özgür olması
- Güvenilir ve güçlü
- Esnek
- Teknik bilgi sahibi olduktan sonra yönetilmesi ve düzenlenmesi için sınırların bulunmaması
- Düşük sahip olma maliyeti (TCO)
- Herhangi bir şirkete bağımlılığın olmaması
- Sahipli donanım ve yazılımların belirlediği kısıtlar olmadan yeni özellik ekleyebilirsiniz

## Linux çekirdeği nerelerde kullanılıyor?

Kısaca *her yerde* kullanılıyor.

- Sunucular
- Süper bilgisayarlar
- Kişisel bilgisayarlar
- Cep telefonları
- Tabletler
- E-kitap okuyucular
- Akıllı ocaklar
- Saatler
- Video oynatıcılar
- ADSL Modemler
- Apple PowerPC



## Linux çekirdeği nerelerde kullanılıyor?

Şu anda nerede ise tüm üniversite bilgi işlem merkezlerinin yanı sıra, NASA, IBM, HP, Boeing, HSBC, CityBank, SonyEricsson, Nokia, Siemens, Samsung, General Motors, Hyundai, Oracle, EToys gibi çok büyük kuruluşlar Linux kullanmakta ve bir şekilde özgür yazılımlara destek olmaktadır.

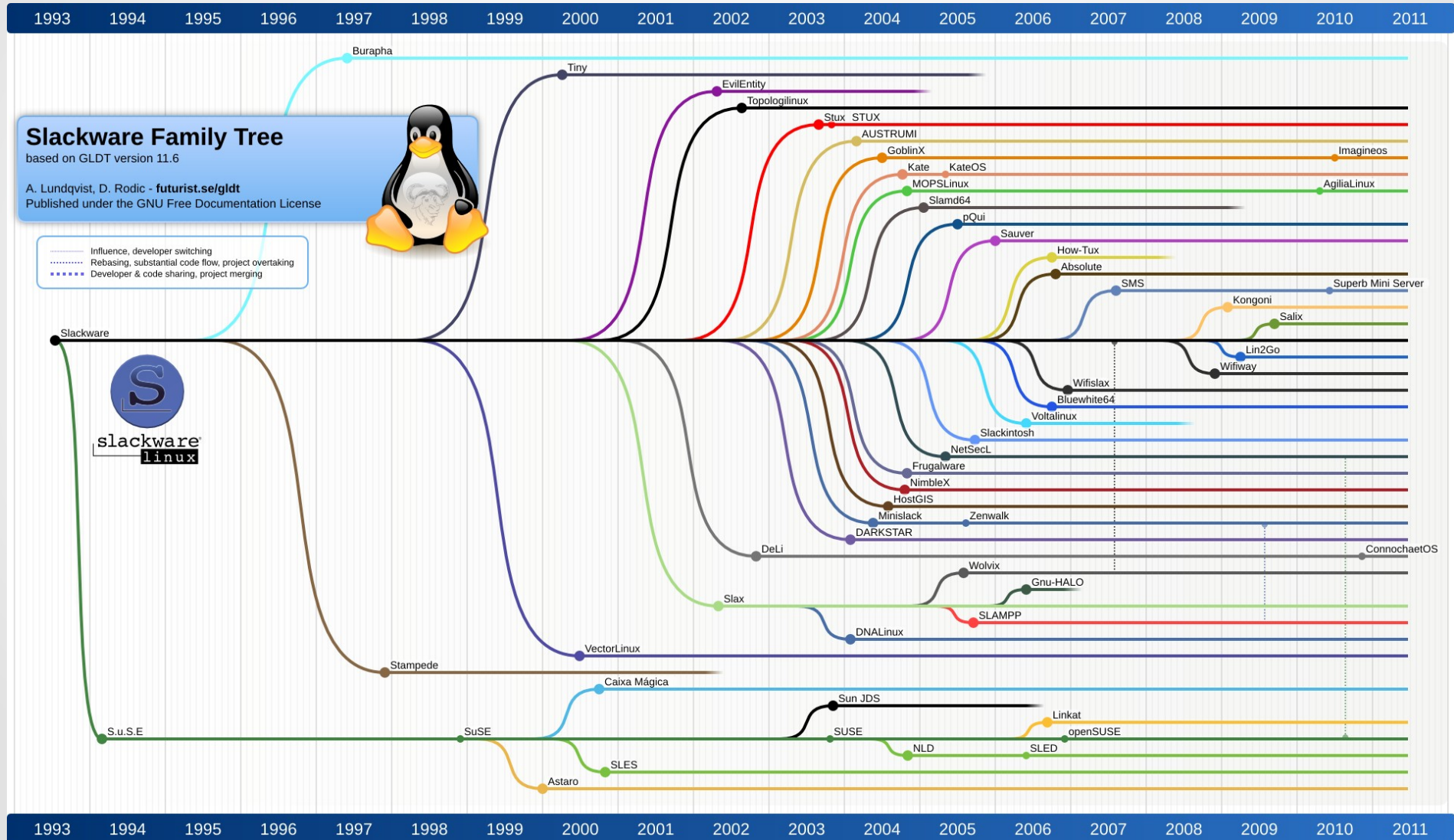


## GNU/Linux Dağıtımı Nedir?

Linux çekirdeği, GNU araçları ve bir takım araçların bir araya gelmesiyle, bu birlikteliği sürdürülebilir şekilde yönetecek bir yapılandırma araçları seti, yazılım güncelleme araçları vb. ile oluşturularak tam teşekküllü bir işletim sistemi haline gelen uygulamalar bütünüdür.



# GNU/Linux Dağıtımları Nelerdir?



## GNU/Linux Dağıtımları Nelerdir?

### Özetle en çok kullanılan dağıtımlar:

- *Debian*, Ubuntu, Linux Mint, **PARDUS**
- RedHat, Fedora, *CentOS*
- OpenSUSE
- Arch Linux, Manjaro, Antergos

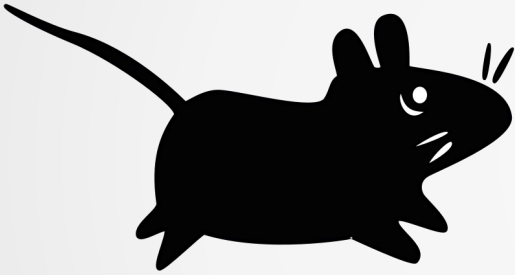
### Kullanıma göre özelleşmiş dağıtımlar:

- Pfsense (firewall sunucu)
- FreeNAS (NAS dosya sunucusu)
- PacketFence (NAC sunucusu)

Page Hit Ranking		
Data span:		
Last 6 months		Go
Rank	Distribution	HPD*
1	<a href="#">Mint</a>	2673▲
2	<a href="#">Manjaro</a>	1914▲
3	<a href="#">Debian</a>	1652▼
4	<a href="#">Ubuntu</a>	1545▲
5	<a href="#">Antergos</a>	1243▼
6	<a href="#">Solus</a>	1199▲
7	<a href="#">Fedora</a>	925▲
8	<a href="#">elementary</a>	924▲
9	<a href="#">TrueOS</a>	881▲
10	<a href="#">openSUSE</a>	857▼
11	<a href="#">Zorin</a>	679▬
12	<a href="#">CentOS</a>	605▲
13	<a href="#">Arch</a>	578▲
14	<a href="#">antiX</a>	558▲
15	<a href="#">Kali</a>	546▬
16	<a href="#">MX Linux</a>	543▲
17	<a href="#">deepin</a>	527▼
18	<a href="#">ReactOS</a>	510▬
19	<a href="#">PCLinuxOS</a>	504▼
20	<a href="#">Lite</a>	464▬

<https://distrowatch.com/>

## Masaüstü Ortamları



# Masaüstü Ortamları ve Pencere Yöneticileri

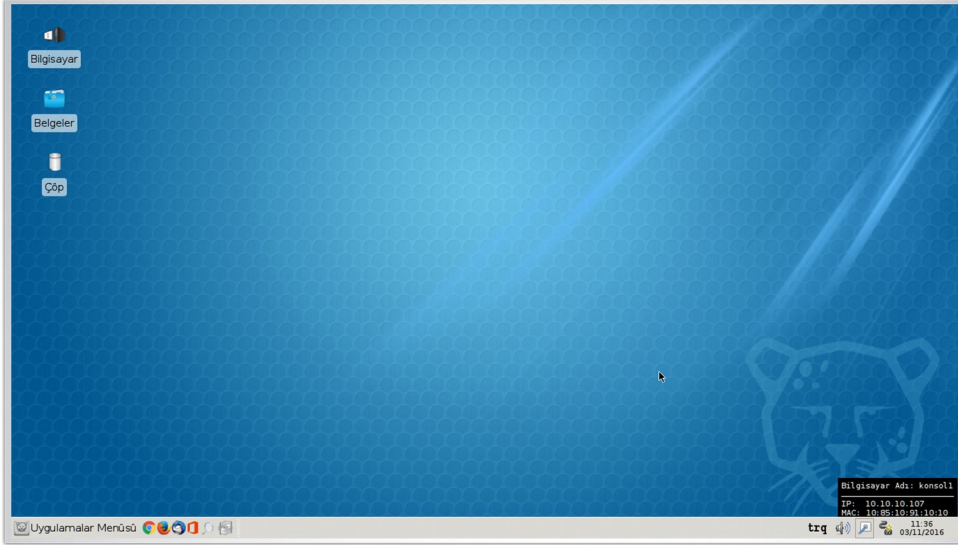
V·T·E Desktop environments, widget toolkits, and window managers based on the X Window System or Wayland		
Desktop environments (comparison)	GTK-based	Budgie · Cinnamon · GNOME · GPE · LXDE · MATE · ROX Desktop · Sugar · Xfce
	Qt-based	KDE Plasma · Lumina · LXQt · MoonLightDE · OPIE · Razor-qt · Trinity
	Motif-based	CDE · IRIX Interactive Desktop · VUE
	Other	EDE (FLTK-based) · Étoilé (GNUstep-based) · Mezzo · OpenWindows (OLIT-based) · UDE (Xlib/XCB-based) · Enlightenment (EFL-based)
Graphical shells	GNOME Shell · KDE Plasma 4 · KDE Plasma 5 · Maynard · Unity	
X window managers (comparison)	Compositing	Compiz · KWin · Metacity · Muffin · Mutter · Xfwm · Enlightenment
	Stacking	4Dwm · 9wm · AfterStep · amiw · Blackbox · CTWM · cwm · Fluxbox · FLWM · FVWM · IceWM · JWM · Matchbox · Motif Window Manager · olwm · Openbox · Qvwm · Sawfish · swm · twm · tvwm · vwm · Window Maker · WindowLab · wm2
	Tiling	awesome · bspwm · dwm · herbstluftwm · i3 · Ion · larswm · ratpoison · StumpWM · wmii · xmonad
Wayland compositors	Enlightenment · Gala · KWin · Mutter · Muffin · Budgie-wm · Weston	
Related	Display server · GUI · GUI widget · Painter's algorithm · Resolution independence · Shell · Virtual desktop · Widget toolkit · WIMP · Windowing system	



## Masaüstü Ortamları

### XFCE

Daha az bellek kullanmakta



### KDE

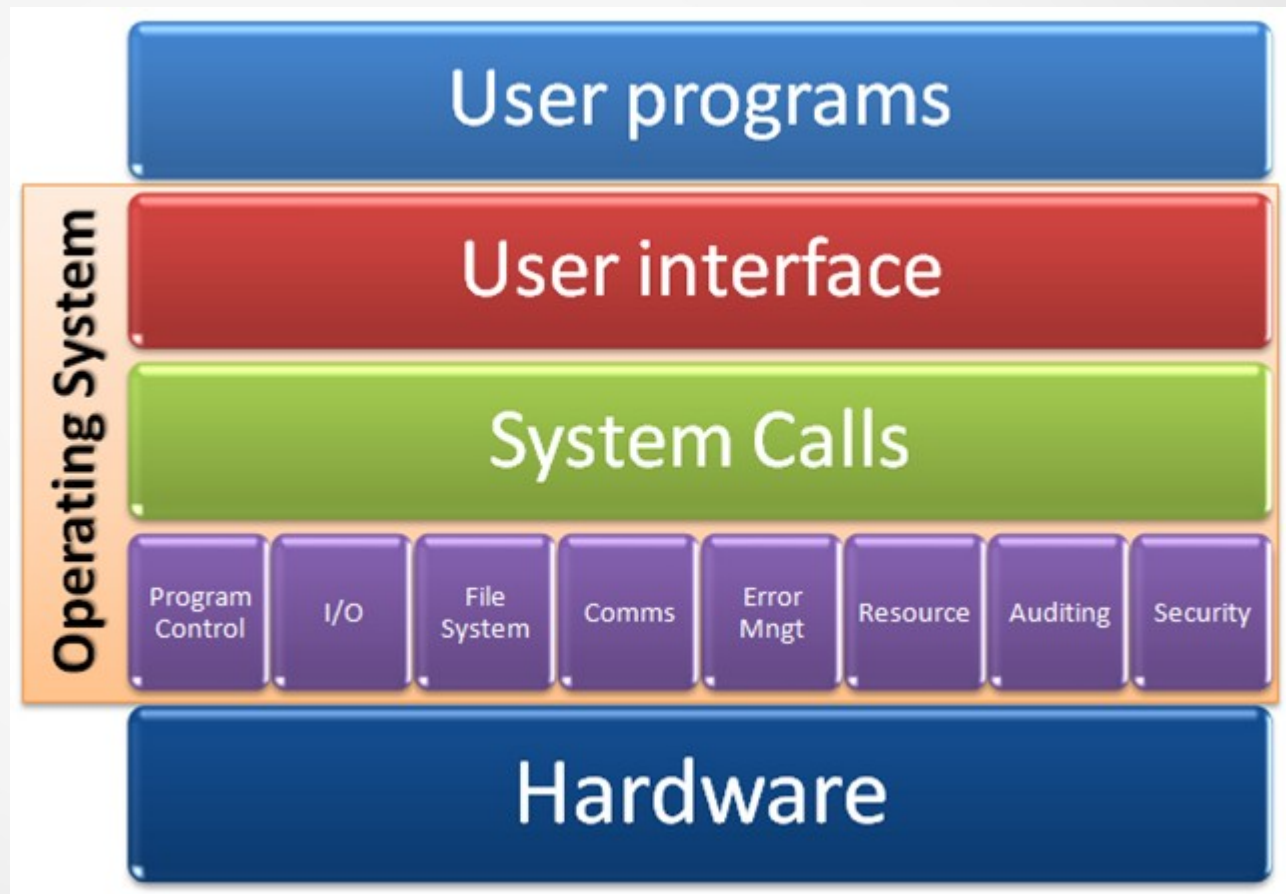
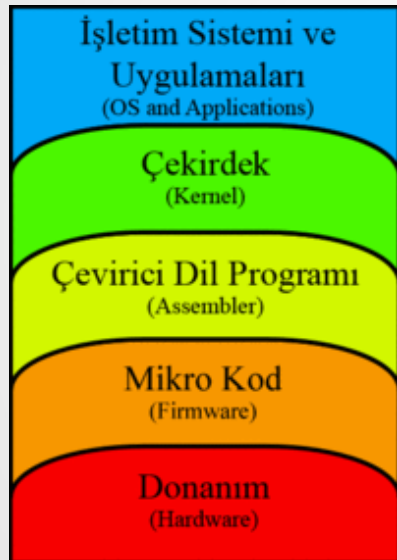
Daha çok görsellik içermekte



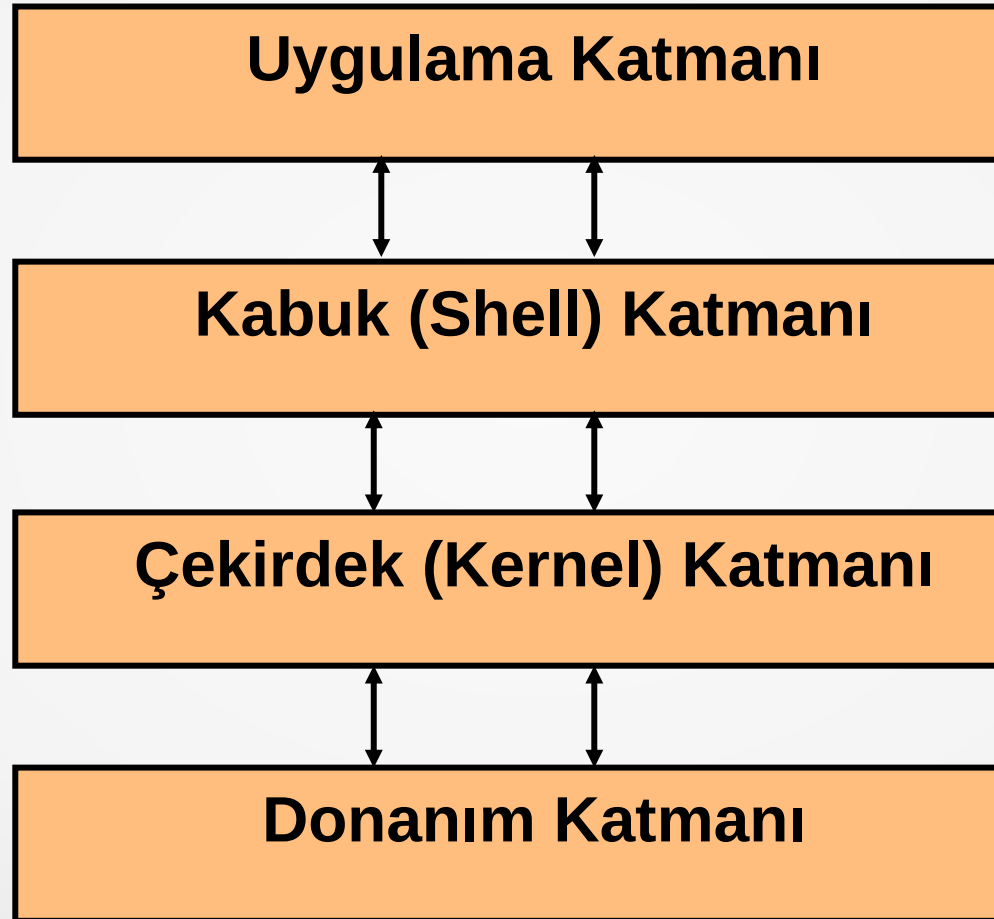
### XFCE

Pardus 17.3 Sürümü

# Temel Bilgisayar Mimarisi

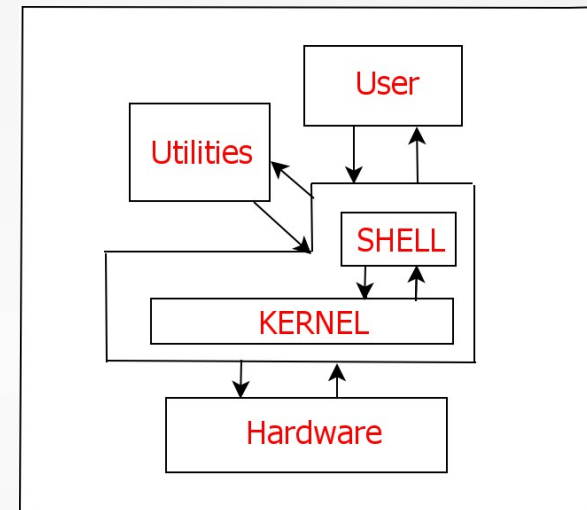
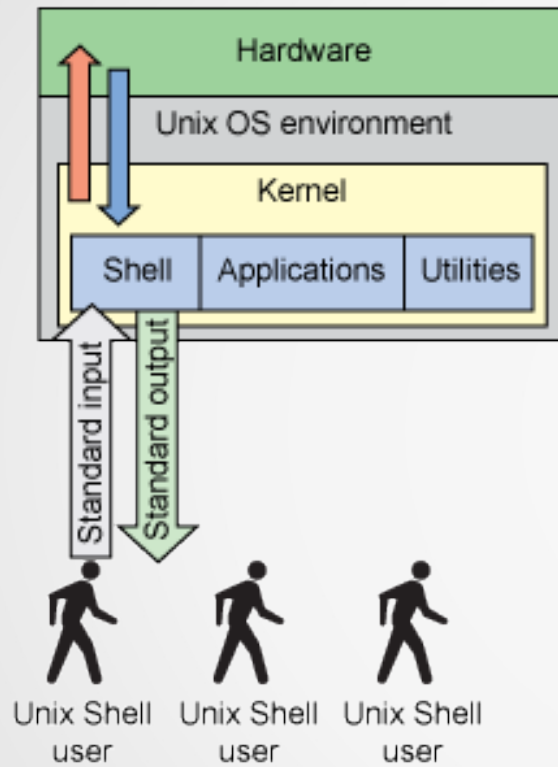


## Özetle İşletim Sistemi Katmanları





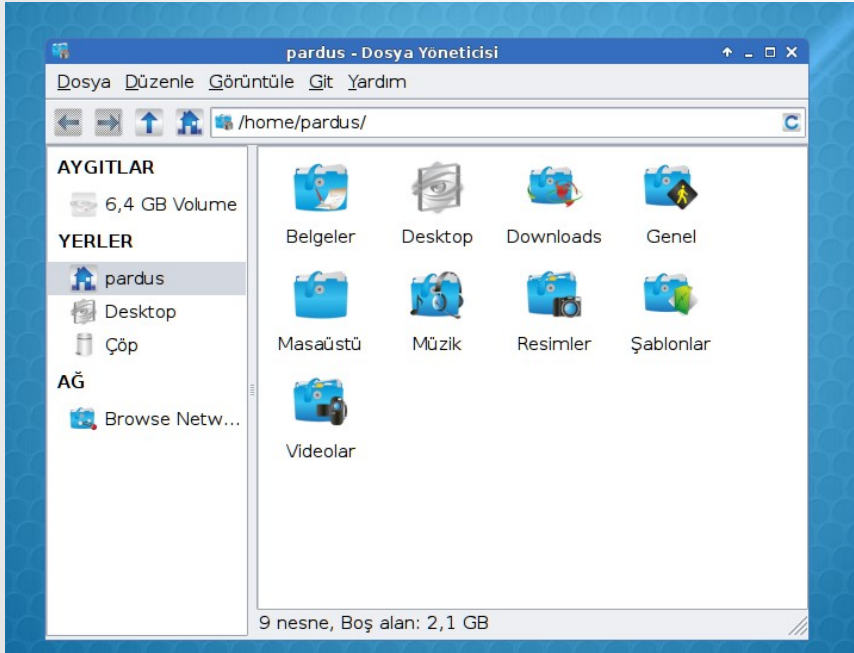
## Kabuk Nedir?



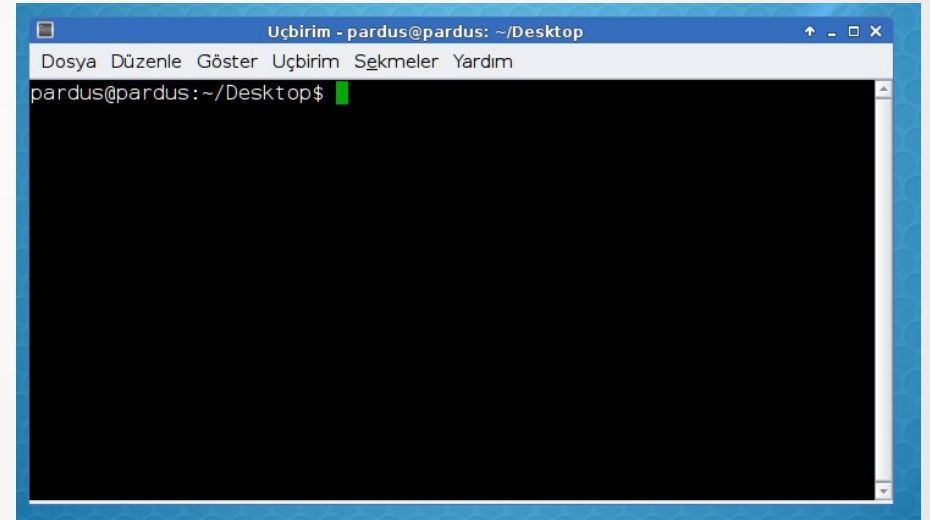
**Shell – Kabuk**  
**Terminal – Uçbirim**

# Dosya Yöneticisi

## Dosya Yöneticisi Pardus XFCE üzerinde "Thunar"



## Uçbirim (konsol, terminal) Pardus XFCE üzerinde



## Sanallaştırma Kavramı

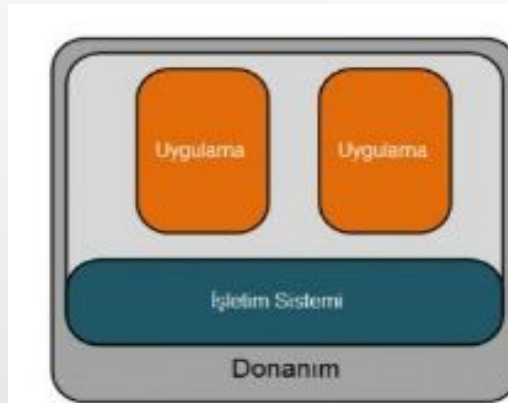
Sanallaştırma, işletim sistemleri ve donanımların herhangi birisinin aslını kullanmak yerine sanallaştırılmış olarak kullanmaktır.

Bu sayede bir işletim sistemi üzerinde aynı donanımlarla birden fazla işletim sistemi çalıştırılabilmektedir.

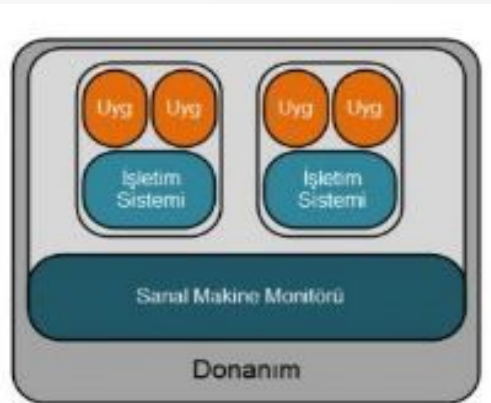
Bu teknoloji yönetim maliyetleri ve işlemci gücünden tasarruf sağlamaktadır.

2005 yılından itibaren sanallaştırma uygulamaları hayatımızda önemli yer elde etti.

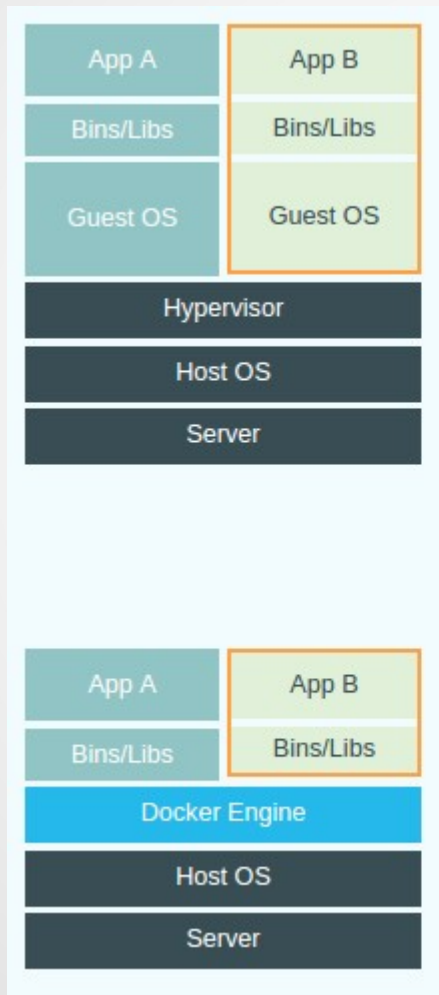
**Sanallaştırma olmadan**



**Sanallaştırma ile**



## Daha da ilerisi Docker



Sanal Makine Mimarisi

Docker Mimarisi

## Kullanılan Kaynaklar

- <http://www.linuxfocus.org/Turkce/March1998/article27.html>
- <http://ab.org.tr/ab12/sunum/202.pdf>
- <https://seminer.linux.org.tr/wp-content/uploads/ozguryazilimlisanslari.pdf>
- <http://ozgurlisanslar.org.tr/>
- <http://www.nyucel.com/2017/05/ozgur-yazlm-ile-ack-kaynagn-ne-fark-var.html>
- <http://www.nyucel.com/2017/05/ozgur-yazlm-m-ack-kaynak-m-1.html>